RADC-TR-76-394, Volumes I - III (of eight)
Final Technical Report
January 1977

COMMUNICATIONS PROCESSOR SYSTEM

North Electric Company

D D C
MAR 14 1977

ROME AIR DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
GRIFFISS AIR FORCE BASE, NEW YORK 13441

Because of the small size of each volume, volumes have been combined into the following reports: Volumes I - III, Volumes IV & V, Volumes VI & VII and Volume VIII.

This report has been reviewed by the RADC Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

This report has been reviewed and approved for publication.

APPROVED:

DANIEL J. McAULIFFE
Project Engineer

APPROVED:

FRED I. DIAMOND
Technical Director
Communications & Control Division

FOR THE COMMANDER:

JOHN P. HUSS
Acting Chief, Plans Office

Do not return this copy. Retain or destroy.

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RADC-TR-76-394, Vols I - III (of eight)	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* COMMUNICATIONS PROCESSOR SYSTEM		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report June 1973 - March 1976
		6. PERFORMING ORG. REPORT NUMBER N/A
7. AUTHOR(s) Mr. Kenneth Hagstrom Dr. Boris Beizer, Data Systems Analysts		8. CONTRACT OR GRANT NUMBER(s) F30602-73-C-0314
9. PERFORMING ORGANIZATION NAME AND ADDRESS North Electric Company 553 South Market Street Galion OH 44833		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62702F 45194904
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center (DCLT) Griffiss AFB NY 13441		12. REPORT DATE January 1977
		13. NUMBER OF PAGES 259
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)* Same		15. SECURITY CLASS. *(of this report)* UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*
Same

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Communications Switching, Communications Processors, Processor Architecture, Circuit Switching, Message Switching, Packet Switching, Base Distribution

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This report covers the results of a study to develop a hardware architecture which will be the basis for a family of communications processors for applications processors for application in circuit, message, packet and base communications switch configurations. Over 23 switching equipments were investigated from which a functional baseline was defined for use in the subsequent studies for evolving an advanced Communications Processor System (CPS) architecture. These switches included circuit, message, and packet switching

DD ¹ FORM 73 **1473** EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

equipments which were felt to be typical of traditional and advanced switching concepts. As an example, the AN/TTC-39, AUTODIN, ARPA Network were used as part of the circuit, message and packet switching baselines respectively. Air Force Base Communications studies were used as the baseline in that area.

From this baseline, a set of fifteen primitive functions were derived which represent the needed capabilities for any generally applied communications processor (CP).

The latest in the state-of-the-art in ADP technology was investigated to determine the best and most viable approach to the CPS. The goal being to develop a family of CP's which could be used to satisfy switching needs for circuit, message, packet, base communications applications or in an integrated node of the future.

The results of the investigation were continually the subject of trade-offs through the use of an analytical modelling technique. The final outcome of this effort is a ten part specification detailing the performance requirements of each unit comprising the communications processor.

This report is organized into eight volumes as follows: Volume I - Executive Summary; Volume II - Definition of Problem; Volume III - Modelling; Volume IV - CCC Architecture; Volume V - CPS Architecture; Volume VI - Software Generation; Volume VII - Appendices; and Volume VIII - CPS Central Processor Specification.

VOLUME I

EXECUTIVE SUMMARY

TABLE OF CONTENTS

TABLE OF CONTENTS - Continued

LIST OF FIGURES

EVALUATION

This report documents the findings of nearly three years of study
into the optimum design for a communications processor. Described in this
report is a highly flexible processor architecture which will be the
basis for the control structure in an Integrated Circuit & Packet Switch
(ICAPS) under TPO IV. The results of this effort will be further in-
vestigated and merged with related software efforts to eventually produce
a validation model of an ICAPS.

APPROVED: *Daniel J. McAuliffe*

DANIEL J. McAULIFFE
Project Engineer

1.0 PURPOSE

This Executive Summary is intended to provide a capsule over-view of the objectives, technical requirements, conduct and re-sults of the Communications Processor System (CPS) program as presented in this final technical report as well as development recommendations for subsequent phases of the CPS program. The term "Final Technical Report" may be somewhat of a misnomer since this report covers only phase one of a program that is presently envisioned as a multi-phased effort. The intent of phase one was to establish a basic Central Computing Complex (CCC) architecture based upon an analysis of existing Message and Circuit Switching systems requirements. The selected architecture is described in Volume IV of this report as well as in the Configuration Item Development Specification which was a separate data item require-ment of this program.

During the course of this program, approximately 30 months, many volumes of reports, and analysis dealing with the many and diverse systems requirements have been prepared and distributed to a limited number of personnel. It was our intent in this final report to summarize the various documents and to present one documentation set which satisfies the SOW data requirement and provides a baseline document to which succeeding phases of the program can be referenced.
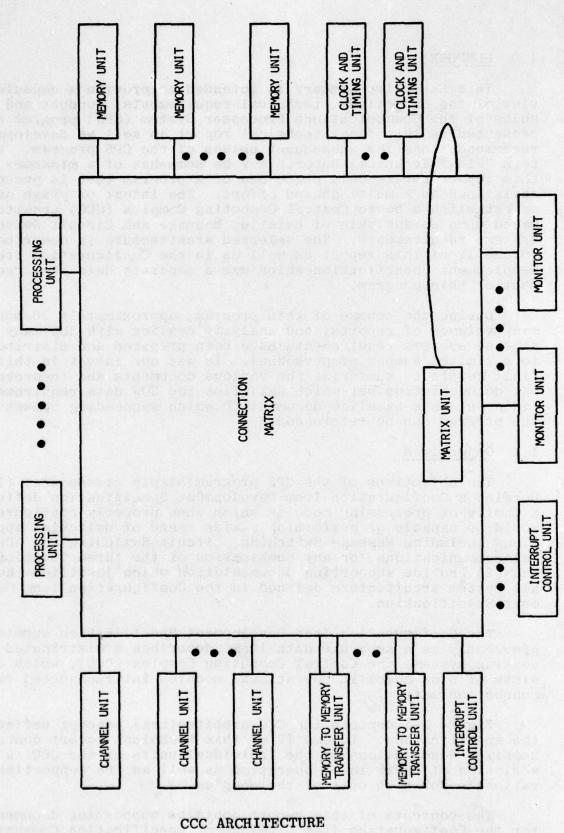
1.1 Objectives

The objectives of the CPS program simply stated are; (1) Develop a Configuration Item Development Specification defining a family of processing modules which when properly configured would be capable of performing a wide range of switching applica-tions including Message Switching, Circuit Switching and ADP Telecommunications, or any combination of the three functions; and (2) Provide supporting documentation which justifies the over-all system architecture defined in the Configuration Item Develop-ment Specification.

The Configuration Item Development Specification submitted previously as a separate data item, describes a distributed pro-cessing system, the Central Computing Complex (CCC), which con-sists of nine specific operational modules interconnected via a connection matrix.

Figure 1-1 depicts the CCC architectural concept defined in the specification. Volume IV of this technical report contains detailed descriptions of the individual units of the CCC, a de-scription of their inter-operation as well as the supporting rationale for each unit of the complex.

The contents of this report contains supporting documentation for the Configuration Item Development Specification Communications

CCC ARCHITECTURE

FIGURE 1-1

I-2

Processor System. It includes:

(1) Definition of the Problems (Volume II)

(2) Modeling (Volume III)

(3) CCC Architecture (Volume IV)

(4) System Architectures (Volume V)

(5) Software and System Generation Techniques (Volume VI)

(6) Appendices (Volume VII)

1.2 Technical Requirements

The major technical requirements specified by the CPS Study Statement of Work are as follows:

Task 1: Establish an application baseline from which to formulate a system architecture and programming technique.

Task 2: Derive *performance* and configuration (modularity) characteristics of the CPS based upon the Application Baseline studies.

Task 3: Develop a Communications Processor (CP) architecture which satisfies the functional requirements of a Communications Processing System.

Task 4: Perform modeling of the selected CP architecture to evaluate the effect of varying such parameters as; memory speed and size, traffic load variations, etc.

Task 5: Analyze System Generation Techniques required for producing and utilizing the entire on-line operational software package.

Task 6: Produce a logic block diagram of the CCC architecture.

Tasks 1 and 2 are covered in Volume II.

Task 3 is covered in Volumes V and VI.

Task 4 is covered in Volume III.

Task 5 is covered in Volume VI.

Task 6 is covered in Volume V.

I-3
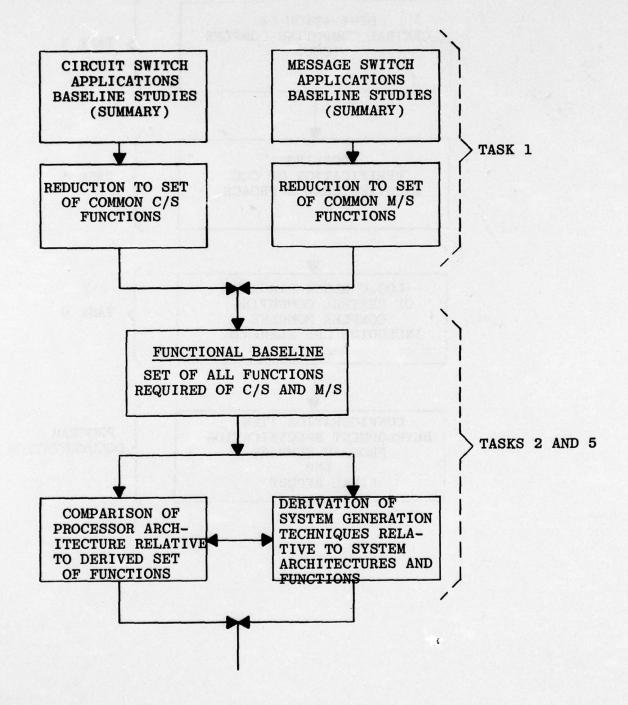
1.3 Program Conduct

The program was organized and conducted to follow a logical flow from the study of existing or planned systems to the derivation of the CCC described in the Configuration Item Development Specification. A generalized program flow chart is depicted in Figure 1-2 and 1-2A. This figure represents major identifiable program milestones. It should not be inferred from the figure that the result of this program as an "ideal" architecture arrived at deductively. On the contrary, the state-of-the-art of communication system design has not yet progressed to the point where it is a "cookbook" process allowing deductive methods to result in an "ideal" system. Communication system design is, rather, a procedure which requires, on the part of the designer, a great deal of experience, a thorough knowledge and understanding of the problem, and the liberal application of equal parts of ingenuity and inspiration to produce an efficient and cost effective system.
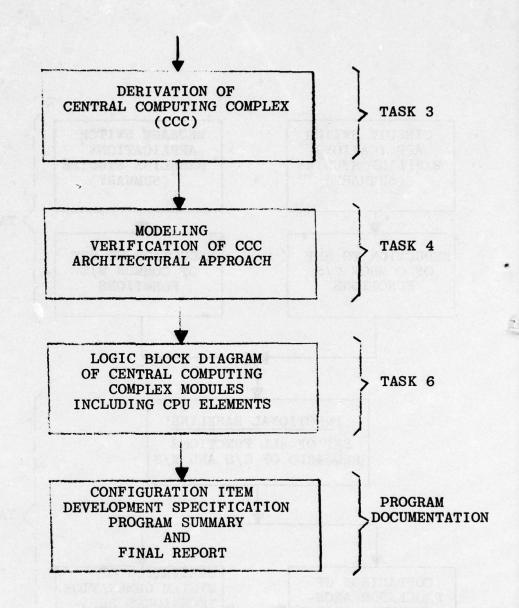
The deductive process can be used, as it was in this program, to define the overall problem in general terms and reduce it to a small set of specific problems. In this case, the overall problem was reduced to a set of fifteen general functions whose properties are such that any system capable of performing all fifteen functions can perform all three of the communication missions, circuit switching, message switching, and packet switching.

This reduction, to a set of distinct functions, was necessary to allow various processor architectures to be applied to the problem in a manageable manner to determine which were best suited to providing solutions. It also allowed the Central Processor (CP) characteristics and instruction repertoire to be defined in great detail, thereby, minimizing the specific questions which were required to be modeled.

Initially the program was segmented into two separate areas, Message Switching and Circuit Switching. In this context, message switching includes Store-and-Forward message switching, packet switching, and base distribution. Each area was required to answer one basic question; "What are the functions performed by operational Message or Circuit Switching systems presently being utilized or planned by the Government?" Answering this question permitted the functions to be combined into a definitive functional baseline document from which the remainder of the program could be based. The combined functions were then compared to existing classical processor system and unit architectures to determine which one, if any, best suited the basic requirements of the CPS functional baseline. This analysis resulted in the basic system architectural concept depicted in Figure 1-1. Up to this point in the program, all system decisions had been made based upon individual system designer's judgement and intuition. In order to verify the decision making process and to evaluate the effects of varying system operational parameters, the

```
┌─────────────────────┐              ┌─────────────────────┐
│   CIRCUIT SWITCH     │              │   MESSAGE SWITCH    │
│    APPLICATIONS      │              │    APPLICATIONS     │
│  BASELINE STUDIES    │              │  BASELINE STUDIES   │
│     (SUMMARY)        │              │     (SUMMARY)       │              ╲
└─────────────────────┘              └─────────────────────┘               ╲
           │                                    │                            ⟩  TASK 1
           ▼                                    ▼                           ╱
┌─────────────────────┐              ┌─────────────────────┐
│  REDUCTION TO SET    │              │  REDUCTION TO SET   │
│  OF COMMON C/S       │              │  OF COMMON M/S      │
│  FUNCTIONS           │              │  FUNCTIONS          │
└─────────────────────┘              └─────────────────────┘
           │                                    │
           └────────────────┬───────────────────┘
                            ▼
              ┌─────────────────────────────┐
              │   FUNCTIONAL BASELINE        │
              │   SET OF ALL FUNCTIONS       │
              │   REQUIRED OF C/S AND M/S    │
              └─────────────────────────────┘
                            │
           ┌────────────────┴───────────────────┐
           ▼                                     ▼
┌─────────────────────┐              ┌─────────────────────┐
│  COMPARISON OF      │              │  DERIVATION OF       │
│  PROCESSOR ARCH-    │◄────────────►│  SYSTEM GENERATION   │
│  ITECTURE RELATIVE  │              │  TECHNIQUES RELA-    │
│  TO DERIVED SET     │              │  TIVE TO SYSTEM     │
│  OF FUNCTIONS       │              │  ARCHITECTURES AND   │
└─────────────────────┘              │  FUNCTIONS           │
           │                         └─────────────────────┘
           └────────────────┬───────────────────┘
                            ▼
```

TASKS 2 AND 5

REPORT ORGANIZATION

FIGURE 1-2

I-5

```
              │
              ▼
┌─────────────────────────────────┐
│         DERIVATION OF           │  ┐
│   CENTRAL COMPUTING COMPLEX     │   }  TASK 3
│            (CCC)                │  ┘
└─────────────────────────────────┘
              │
              ▼
┌─────────────────────────────────┐
│           MODELING              │  ┐
│      VERIFICATION OF CCC        │   }  TASK 4
│    ARCHITECTURAL APPROACH       │  ┘
└─────────────────────────────────┘
              │
              ▼
┌─────────────────────────────────┐
│     LOGIC BLOCK DIAGRAM         │  ┐
│    OF CENTRAL COMPUTING         │   }  TASK 6
│      COMPLEX MODULES            │  ┘
│   INCLUDING CPU ELEMENTS        │
└─────────────────────────────────┘
              │
              ▼
┌─────────────────────────────────┐
│     CONFIGURATION ITEM          │  ┐  PROGRAM
│  DEVELOPMENT SPECIFICATION      │   }  DOCUMENTATION
│       PROGRAM SUMMARY           │  ┘
│            AND                  │
│        FINAL REPORT             │
└─────────────────────────────────┘
```

REPORT ORGANIZATION

FIGURE 1-2A

architecture was analyzed using GASM (Generalized Analytical
System Model) as the basic modeling tool. The modeling program
resulted in several refinements in the basic system philosophy
and implementation recommendations. A parallel effort was also
conducted during the entire CPS program relating to system soft-
ware problems and requirements. The results of both studies were
then merged to produce a Configuration Item Development Specifica-
tion defining the overall processing complex architecture. The
Configuration Item Development Specification along with this
final technical report comprise the significant data outputs of
the CPS program.

2.0 REPORT FORMAT

2.1 General

This report has been divided into seven volumes of which the Executive Summary is Volume I. The remainder of the report has been divided to the extent possible into volumes by subject matter, not necessarily by chronological performance order. A brief description of each volume has also been included to provide the reader with a basic overview of the data content of this report in its entirety. One other document which provides additional relevant CPS information is the Configuration Item Development Specification. This document was submitted previously under a separate cover and is available thru the RADC publishing departments.

2.2 Volume Descriptions

2.2.1 Volume II, Definition of Problem

This volume of the CPS report contains the Applications Baseline, and Functional Baseline study results as well as an analysis of system architectures which were evaluated. The Application Baseline was the starting point for all subsequent phases of the study. The baseline information summarized the Message and Circuit Switching Systems studied, detailed the operational and functional requirements of each type of system and provided consolidated sets of requirements on which the CPS program was based. The Functional Baseline study consolidated the Message and Circuit Switching requirements into a single set of functional and operational requirements which encompassed both types of systems. These requirements were then evaluated and compared against various types of system architectures to determine which architecture best suited the combined requirements. The results of this effort are contained in the CCC architectural evolution section.

2.2.1.1 Application Baseline Description

Section 1 of Volume II, Communication System Application Baseline, summarizes the salient features of the Message and Circuit Switching systems analyzed. It is divided into two areas: (1) Circuit switching systems and their control functions, (2) Message Switching Systems and their control functions.

2.2.1.2 Functional Baseline Description

Section 2 of Volume II, Communication System Functional Baseline, derives a CPS Functional Baseline based upon the results of the Application Baseline studies.

2.2.1.3 CCC Architectural Analysis

Section 3 of Volume II discusses the various architectural concepts evaluated as well as providing the rationale for selecting the distributed function processing system finally arrived at.

2.2.2 Volume III, Modeling

This volume contains a description of the overall modeling effort, a description of the Generalized Analytical Systems Model (GASM, the software tool used to perform the actual modeling), the parameters varied during the modeling effort, and the actual results of the modeling.

2.2.3 Volume IV, CCC Architecture

This volume contains a Programmer's Reference Manual which describes the selected CCC architecture, a description of the individual units which comprise the CCC as well as the rationale for including each unit in the complex. The bulk of the document consists of the explanation of the choices, supplementary notes, design details, examination of problems, and trade-off considerations.

2.2.4 Volume V, CPS Architectures

This volume is divided into four sections, each section addressing a specific area of concern.

2.2.4.1 Application Considerations of CPS Architectures

This section shows various system configurations possible utilizing the CCC units described in Volume IV. Included for discussion are: (1) Typical Message Switching systems, (2) Typical Packet Switching systems, (3) Typical Circuit Switching systems, (4) Tech Control Systems, and (5) Digital Concentrators. Functional as well as physical descriptions are included for each type of system contained in this section.

2.2.4.2 Partitioning of CCC Units

This section describes a tentative partitioning of the various units described in Volume IV into LSI chips. This section does not attempt to finalize the chip partitioning but is only intended to be a first-cut partition which will no doubt be modified during subsequent phases of the CPS program.

2.2.4.3 Integrated Circuit Projections

This section presents the results of an industry survey as to what can be expected in the area of chip density for the

various technologies which will be utilized in the CPS development time-frame.

2.2.4.4 Memory Technology Projections

This section addresses the current status of feasible erasable mass memories which could be considered for CPS applications. Included for analysis are: (1) Semiconductor type, (2) Magnetic Bubbles, and (3) Optical Memories.

2.2.5 Volume VI, Software Generation

This volume addresses the overall system generation problem and includes an analysis of possible usage of a "higher level language" as well as sections which address the requirements necessary to identify, select and configure Circuit and Message Switch software over a wide range of functional equipment arrangements.

2.2.6 Volume VII, Appendices

This volume contains appendices which support the information contained in Volumes II through VI. Included are:

Appendix I - Classical Processor Architectures

Appendix II - Use of Content Addressable Memory In Communications Switching

Appendix III - Glossary of Terms and Acronyms

Appendix IV - Bibliography

Appendix V - Circuit Switch and Message Switch Functional Breakdowns and Calculations

3.0 REPORT SUMMARY

An applications baseline study was undertaken in which a wide variety of communications systems, including message switches, circuit switches, packet switches, and base distribution switches, were studied in detail. The systems studied included the entire gamut of communications systems from very small to very large (in terms of lines, trunks, circuits, and terminals), from low traffic per inlet to very high traffic per inlet, and from individual systems to entire interconnected networks.

The guiding documents, which were used to perform this study and which are described in the Bibliography (Volume VII, Appendix IV), were those planning documents relating to the defined current and planned needs for defense communications in the period 1975 to 1985. While many of these planning documents in their current state do not represent in any way specifications for the required systems, they do more than anything else represent the desires and design concepts of what will be needed in the next ten years to meet the growing and broadening needs for communications. With these designs and ambitions in mind, selection of those systems which reflected technology developments that would be ultimately useful and applicable to the needs of the future defense communications systems was made. The purpose of the applications baseline study was to set down the basic underlying concepts regarding contemporary communications systems so that various interested parties could criticize, improve and jointly evolve the functional baseline for the CPS computer family.

Of the hundreds of processor controlled systems now in operation or in planning, only a small fraction could be studied in detail. The selection of systems to be studied, both data and circuit switching, was based on one or more of the following:

(1) The system was representative of a class of similar systems.

(2) It fit directly in with present and projected military communication requirements.

(3) It was, or is expected to be, successful and cost effective.

(4) It did, or will exist, in multiple copies.

(5) It contains some unique features which may have bearing on future system designs.

(6) It represents the current state-of-the-art within a specific switching mission category.

The results of the Applications Baseline Studies were used to generate a Functional Baseline against which the use of various processor architectures could be investigated. Two facts of prime importance emerged from this effort: (1) all of the systems studied employed a distributed processor architecture even those which featured a central processor, and (2) the work performed by circuit and message switching systems could be reduced to a set of fifteen functions whose properties are such that any system capable of performing all fifteen to any degree of complexity could perform any switching mission.

Each function of the set was considered as a specific processing problem which had to be solved individually and, in some cases, in combination with other problems (functions). Various processor unit and system architectures were applied to the problems to determine which, if any, was best suited to providing the solutions.

The results showed, not surprisingly, that no single unit architecture was always superior over all others. The conclusion reached is that future systems will also have a distributed control architecture and that a uni-processor Communications Processor Unit (CPU) architecture is, overall, the most efficient approach.

Of all of the existing systems studied, none remained or will remain static during its service life. Each system had gone (or is going) through modifications to its hardware or software or both to accommodate new or expanded requirements. This leads to a second major conclusion; the Central Processor's (CP's) architecture must facilitate adaptation to new or expanded requirements with a minimum disruption of service, and preferably, be amenable to these changes while "on-line".

Another major conclusion derived from the Functional Baseline Study is that any CP which could fit any of these applications efficiently must be highly flexible in terms of size (the number of units equipped) and capability (the types of units equipped) without changing the basic architecture. When the spectrum of potential applications is examined, it becomes apparent that extreme flexibility in terms of processing capability is required. The CP architecture must be efficient when used in a 300 line circuit switch, where the equivalent of a mini-processor is all that is required on-line, and it must also be equally efficient when used in a 400 termination base distribution message center, where the equivalent of several full-blown processor systems are required on-line.

A fourth major requirement is architectural survival. If a reduction of life cycle costs of hardware and software is to be realized in future communications systems, it is absolutely

mandatory that the central processor architecture be inherently flexible enough to survive through several generations of changes in system requirements and hardware technologies.

The fifth major conclusion is that, for efficient operation, the CP architecture must have the inherent ability to make maximum use of the system's resources and require minimum replication of units, preferably allowing an N+1 redundancy philosophy to be implemented.

The fifteen communications functions provided guidance in defining the operational characteristics of the CP, i.e., the instruction repertoire, I/O requirements, channel requirements, addressing requirements, etc. The five major conclusions cited above provided guidance in evolving the architecture of the CP. The modeling effort (described in Volume III) provided the means to test various concepts and ideas and helped to fine-tune the architectural concepts and CP characteristics.

Finally, the architecture of the CP emerged. It was soon realized that the CP was more of a computing complex than a central processor so the term Central Computing Complex (CCC) was adopted. This is the term used throughout the remainder of this report.

The entire CCC concept, which is described in detail in Volume IV, is based on the assumptions concerning Integrated Circuit (IC) technology contained in the CPS study Statement of Work.

It is assumed that an IC technology will (or does exist) which will result in gate densities on the order of 2000 to 5000 per 150 mil square chip with a quiescent power dissipation of 1 milliwatt per chip (200 nW per gate). It is also assumed that the on-chip gate delays of this technology will be on the order of 3 to 5 nanoseconds. The technology projections described in Volume V, Section 3 of this report indicate that these assumptions are not unrealistic (refer specifically to the I^2L technology description).

The CCC is a distributed system architecture composed of semi-autonomous units. Advantage has been taken of large scale integrated circuit technology to make the units "smart". Many of the capabilities normally associated with a CPU have been redistributed and allocated to other units. The heart of the architecture is an interconnection matrix which allows all compatible units to communicate with one another. The architecture has the following characteristics:

    (1) Modularity at the unit level

(2) Replication as needed at the unit level

(3) Automatic graceful degradation and recovery

(4) Self-failure detection

(5) Built-in diagnostic facilities and commands

(6) Built-in monitoring capabilities.

(7) Expandability from a minimum set of units (typically 12 or 13) to a maximum of 254 units.

A unit is the basic interchangeable, replicable element of the architecture. Examples of units are:

Processing Unit (CPU)

The processing unit is a general purpose digital computer. It has a 32-bit word length, character, half-word, and word addressing. The repertoire has been optimized for the communication task. It is a generalized register machine, having 16 registers of 32-bits each. Each register can be split in half and used independently. The machine is paged, with page size of 65,536 characters.

Interrupt Control Unit (ICU)

All interrupts generated in the system are transmitted to the Interrupt Control Unit(s) through the matrix. In addition to the normal interrupts (e.g., transfer termination, special characters, etc.), the ICU gathers, interprets, and acts upon all alarm conditions.

General Purpose Channel Unit (CU)

All devices are connected to the complex by means of a general purpose Channel Unit. All channels provide the following capabilities:

(1) Programmable speed

(2) Master or slave mode

(3) Direct memory transfers

(4) Chained I/O operation

(5) Full-duplex operation.

A channel may be used to terminate more than one physical

I-14

device if a device controller of the proper type has been designed. The device side of the interface is a character interface. The matrix side of the interface is logically a word interface. Tne device side interface is universal. All controllers must be built to serve that interface.

Memory Unit (MU)

The Memory Unit is a module of 65,536 characters. From the programmer's point of view, word, half-word, character, and bit addressing are provided.

Memory-to-Memory Transfer Units (MMTU)

The Memory-to-Memory Transfer Unit is used to make memory-to-memory transfers and other transfers between passive units. Its operation is transparent to the programmer. Each MMTU takes up two unit numbers.

System Monitor Unit (SMU)

The System Monitor Unit is a watchdog unit over the CPU's. It expects signals from the CPU's on a periodic basis. Should an expected signal fail to arrive on time, the monitor unit will initiate corrective action, ranging from reinitializing the faulty CPU to throwing it out of the complex and reassigning a standby CPU to the task.

System Clock Unit (SCU)

The System Clock Unit provides a master timing source and time of day clock for the entire complex. It is addressable and programmable.

Bootstrap Unit (BU)

A unit which can, under operator control, perform the loading of various control memories and unit ID's as necessary to get the system into operation.

Performance Monitor Unit (PMU)

The Performance Monitor Unit is a unit which is equipped with high impedance probes used to monitor activities within the complex. It is used to gather statistics on the complex's performance and internal matrix traffic. Its primary use is for system tuning although it can be helpful in software diagnosis.

Matrix Unit (XU)

The matrix is a distributed system containing a number of

submatrices. Under normal operation, there is no need to explicitly access a submatrix or its controls. However, in the event of a submatrix failure, submatrix restoral, system bootstrap, or other conditions described elsewhere, it is necessary to force the state of the matrix and/or to query its status. For this reason, each submatrix control is also accessible as a unit through data paths that traverse the matrix itself.

The matrix is a distributed, modular structure with inherent graceful degradation capabilities. It is the means by which data and control paths are established between all units of the complex. The operation of the matrix is for the most part transparent to the programmer. The matrix is the functional counterpart of the various busses found in previous system architectures. The matrix control is performed by a distributed control system and shares the graceful degradation capabilities of the matrix proper. Failure of part of the matrix may cause the system to slow down, but will not cause it to fail. Requests for connections are not directly under program control but are derived from the nature of the transfer or instruction being executed. The matrix is self-diagnosing and creates interrupts in the event of improper connection attempts.

All units have the following characteristics:

(1) Common interface with the matrix and with each other

(2) Unit generic micro-instructions

(3) Dynamically modifiable logical identity which can differ from its physical identity.

(4) Interchangeability with any other unit of the same type.

(5) Self-failure and malfunction detection

(6) Interconnection to other units via the matrix

(7) Logically isolatable from other units

(8) Power isolation from other units.

Communication between units and control over units is obtained through the use of a unit level micro-instruction repertoire. There is a substantial amount of overlap between the unit repertoire and the I/O repertoire of the system; in many cases, the commands are indistinguishable. Unit level commands fall into two categories: generic unit commands and unit specific commands. Generic unit commands are identical for all units and can be applied to all units. The behavior of the

unit under a generic command will be the same (except for some details) for all units. A given OPCODE in the generic repertoire can be expected to have the same action in all units. A unit specific command has an interpretation which is peculiar to the unit. The same OPCODE presented to some other unit could result in totally different actions.

In summary, a Central Computing Complex architecture has been derived which satisfies all five of the major conclusions reached as a result of the Functional Baseline Study and which is capable of performing all fifteen of the communications functions regardless of their individual complexity.

4.0 DEVELOPMENT RECOMMENDATIONS

4.1 General

This section contains a tentative plan for the development of the CCC hardware. The earlier phases of the development plan are given in greater detail than the later phases. Also included are recommendations for the tasks to be accomplished in the next phase of the CPS program. A development cycle lasting a minimum of three years (at high cost) and a likely time of four years is envisioned. The reasons for the duration of the development and the expected man power is intimately tied to the peculiarities of large scale integration which is recommended for the next CPS development phase. LSI has a profound effect on the economics of computers. It would be unrealistic to expect that qualitative changes in the development procedure and costs would <u>not</u> occur.

The advent of LSI technology provides a significant qualitative change in the economics of computer hardware: cost is effectively divorced from logical complexity, allowing the economic implementation of units and devices which would have been impractical with MSI or discrete components. LSI, however, has an equally significant impact on the design process and the implementation time and cost. The impact of LSI is to: (1) shift engineering labor from the end of the development cycle to the beginning of the development, and (2) to increase the capital investment required to implement a new line of hardware.

While this may be dismaying, it should not be unexpected. It is representative of a process which has been going on since the beginnings of the industrial revolution and which can be traced further back into antiquity. There is a functional principle in operation: low unit cost is obtained only by capital investment-the lower the target unit cost, the higher that investment. LSI, whatever else it may be, is first and foremost a means of reducing unit cost - that is, it is a production technique. Its value is not so much in reduced wiring or reduced components, but in reduced labor content. The soldering iron operation is replaced with a printing operation.

The net result of LSI, or any form of production automation, whatever the product, is a total reduction in the labor content of the unit; bought at the price of increased capital investment. That investment is not an abstract thing, ultimately it is dominated by labor also, engineering labor rather than technician and production labor; but labor nevertheless. The result of LSI implementation would appear to be a sudden increase in engineering man hours required to design and produce a computer. However, it is a shifting process which has been going on for some time. It was evident in early vacuum tube machines, was more obvious when we shifted to transistors; more so with MSI,

and obvious with LSI. If we conceive of the entire process of getting a computer on the air as starting with initial engineering, prototype construction, production engineering, production testing, fabrication, and final assembly testing, what we find is a gradual displacement of labor content towards the beginning of the process. As each step has been improved in the form of reduced labor, it has been at the expense of increased labor in the previous phase. LSI completes the shift since it redistributes the labor content between initial engineering and production engineering.

Figures 4-1 thru 4-5 constitutes a simplified PERT chart of the development process starting with the conceptual realization of the need for a product, and ending with the acceptance of the first site using the product. While events are represented as discrete activities, no development is ever this simple: there are numerous feedback paths, simultaneous activities, additional interactions, etc., which are not shown in the interest of clarity. Also not shown is the fact that LSI development involves considerable interactions between the manufacturer and the logic designers. Contingencies for rework of special order chips are not shown, having been incorporated into the elapsed times.

This development cycle is considered to take place in six more or less separate phases. The first phase (Product Planning) is described in this section.

Phase 1 - Product Planning

Phase 2 - Development of Internal Design Specifications

Phase 3 - Design

Phase 4 - Unit Testing

Phase 5 - Integration Testing

Phase 6 - Application

4.1.1 Notation

The following notation is used:

(1) Numbers in the circles on the upper right of a box is an identifier of the activity and will be used to coordinate the PERT chart with the narrative description of the next section.

(2) Heavy lines indicate a critical path.

(3) Numbers on the lines indicate the number of months of elapsed time to the conclusion of the next activity. A pessimistic but reasonable number has been used in all cases. That is, this is not a "worst case" PERT chart, nor the "expected" but something in between.

I-19

(4) Numbers in parenthesis on the lines indicate accumu-
     lated time to the earliest completion to the conclu-
     sion of the associated activity.

4.1.2 What Has Not Been Included

We have not included the development of any devices, either
in the PERT chart or in the subsequent costings. The reason for
this is that the specific order in which devices will be designed
and implemented will depend heavily on the first few applications.
Generally, devices will be relatively straightforward and will
not be on any critical path. Device designs should not be initi-
ated prior to month 24 of the development. Device design costs
are expected to be minimal. The following devices (in prototype
form) should be developed regardless of the application, concur-
rently with the development of the CCC:

(1) VDU and Interfaces

(2) Moving Head Disc

(3) Tape Unit

(4) TTY/KBD/PTR and Interfaces
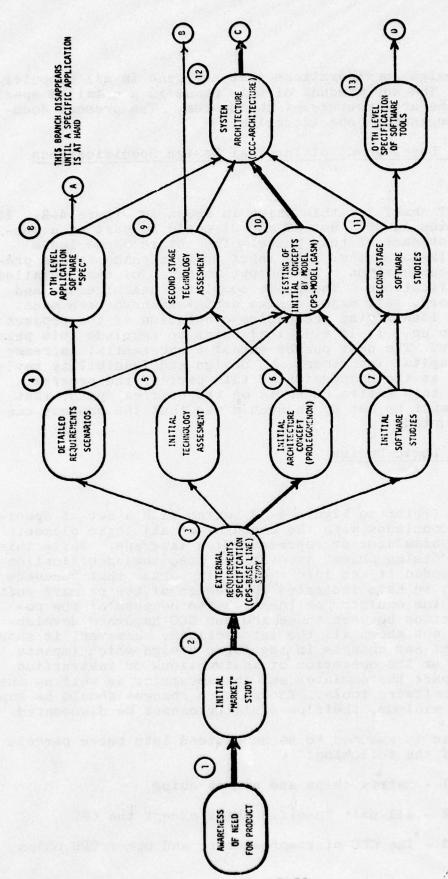
These will be needed for system shakedown testing. How-
ever, the devices which are developed for that purpose need not
be the same as those developed for the CCC applications. What
is needed here are devices which will allow the testing of the
system to go on effectively. We have included the costs to de-
sign such jury rigged devices, probably using MSI rather than LSI.
It should be emphasized that the above devices will not necessarily
correspond to the final versions of these devices as used in a
CCC application.

4.1.3 Conclusions

The overall schedule takes 48 months. A crash effort
could conceivably reduce this to 36 months, at a significant in-
crease in cost and risk. However, equipment in the field, ready
for first application could come as early as month 60 assuming
that the application software and associated design effort had
started early enough. The 48 month development cycle assumes no
hiatus between various phases.

4.2 Phase 1 - Product Planning

This phase is shown in Figure 4-1. It essentially covers
the CCC development to date and can be considered complete in all
significant respects. It has been included for the sake of com-
pleteness and does not influence the timing of subsequent phases.
Generally, it corresponds to the typical market studies and

I-20

FIGURE 4-1 - PERT CHART FOR CCC DEVELOPMENT
PART 1 - PAST TO PRESENT

THIS BRANCH DISAPPEARS
UNTIL A SPECIFIC APPLICATION
IS AT HAND

AWARENESS OF NEED FOR PRODUCT

INITIAL "MARKET" STUDY

EXTERNAL REQUIREMENTS SPECIFICATION (CPS-BASE LINE) STUDY

DETAILED REQUIREMENTS SCENARIOS

INITIAL TECHNOLOGY ASSESMENT

INITIAL ARCHITECTURE CONCEPT (PROLEGOMENON)

INITIAL SOFTWARE STUDIES

0'TH LEVEL APPLICATION SOFTWARE "SPEC"

SECOND STAGE TECHNOLOGY ASSESMENT

TESTING OF INITIAL CONCEPTS BY MODEL (CPS-MODEL,GASM)

SECOND STAGE SOFTWARE STUDIES

SYSTEM ARCHITECTURE (CCC-ARCHITECTURE)

0'TH LEVEL SPECIFICATION OF SOFTWARE TOOLS

I-21

preliminary design considerations that are done in all computer developments. The key product of this phase is a detailed specification for the architecture of the system. The present document constitutes such a specification.

4.3 Phase 2 - Development of Internal Design Specifications

4.3.1 General

The PERT chart for this phase is shown in Figure 4-2. It consists of boxes 14-29. Box 30 and 31 could be part of a subsequent phase or part of this phase. The entire phase lasts approximately 12-15 months. The input to this phase is the present set of documentation. The output is a set of very detailed internal specifications. These are complete, unambiguous, and non-contradictory. All major design decisions have been made. Detailed specs identifying the entire conclusion of the project have been drawn up. It is a logical point to conclude this phase of the activity. The next phases entail a substantial increase in labor and capital investment. A design and feasibility review should be held at the conclusion of this part of the program. However, since this entire phase is on the critical path, that design review must be set up in such a way that the project can go on uninterrupted.

4.4 Phase 3 - Logic Design

4.4.1 General
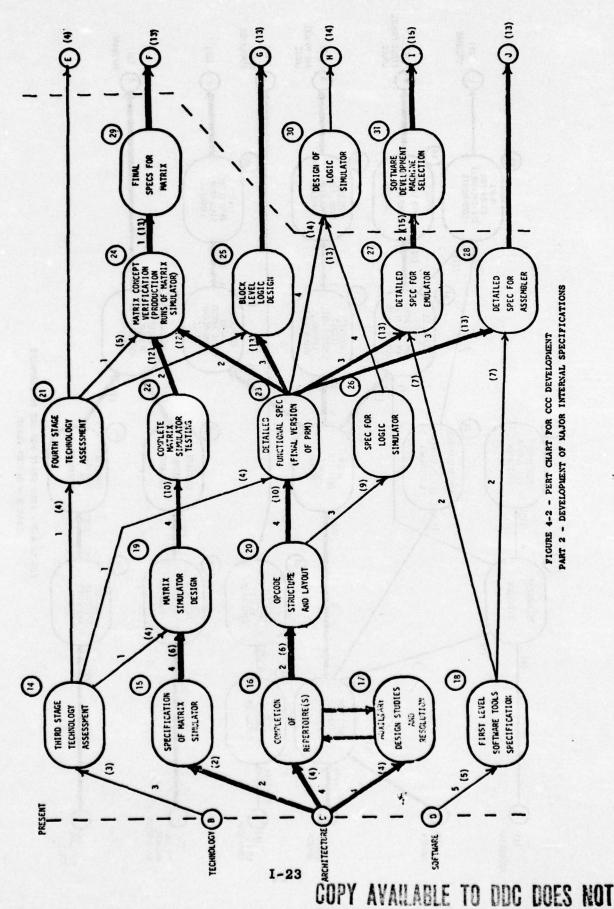
Phase 3 (refer to Figure 4-3) begins with a set of specifications and concludes with the encoding of all logic elements into the logic simulator or representation language. While this is called the "design phase" it overlaps into the specification phase that preceded it and into the testing phase that succeeds it. As before, we have indicated the design of the primary software tools and the emulator on these charts because of the potential interaction between these and the CCC hardware development. We have not shown all the interactions. However, it should be realized that any changes in the logic design which impacts the repertoire or the operation of instructions or instruction modes, will impact the emulator and the assembler as well as the other primary software tools. Though such changes should be kept to an absolute minimum, their possibility cannot be discounted.

The logic is assumed to be subdivided into three parcels that consist of the following:
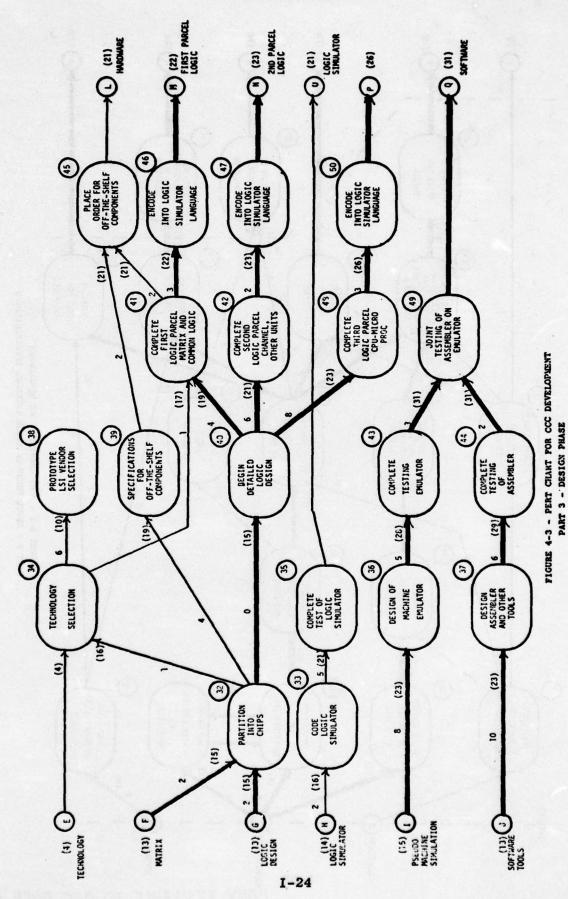
Parcel 1 - Matrix chips and common chips

Parcel 2 - All unit specific chips except the CPU

Parcel 3 - The CPU micro-processor, and other CPU chips.

FIGURE 4-2 – PERT CHART FOR CCC DEVELOPMENT

PART 2 – DEVELOPMENT OF MAJOR INTERNAL SPECIFICATIONS

I-23

FIGURE 4-3 - PERT CHART FOR CCC DEVELOPMENT

PART 3 - DESIGN PHASE

I-24

There is in reality a "Parcel 0" consisting of off-the-shelf chips. We have further assumed for the sake of planning only that the CPU will be implemented as a micro-processor. We should not be obstinate in this. That decision, finally, will be done only when the CCC chip partition, Item 32, is done. If the CPU were to be implemented as logic there would be some changes in the PERT chart whose net effect might be to increase the elapsed time for the hardware completion.

4.5 Phase 4 - Unit Testing

4.5.1 General
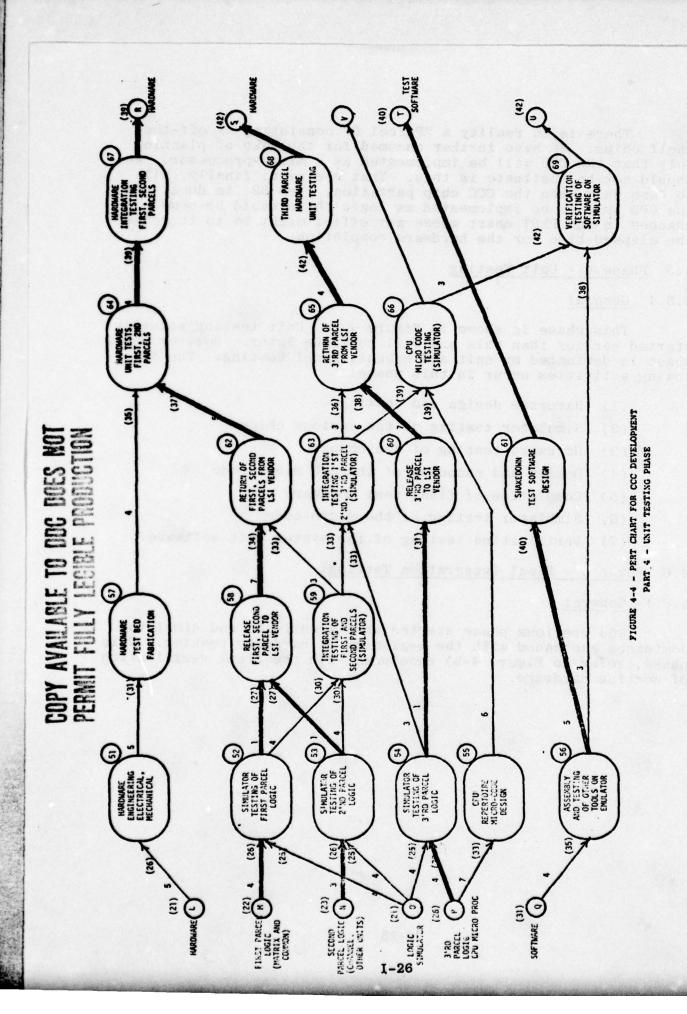
This phase is shown in Figure 4-4. Unit testing actually started earlier than this and will continue later. However, this phase is dominated by unit and subunit level testing. The following activities occur in this phase:
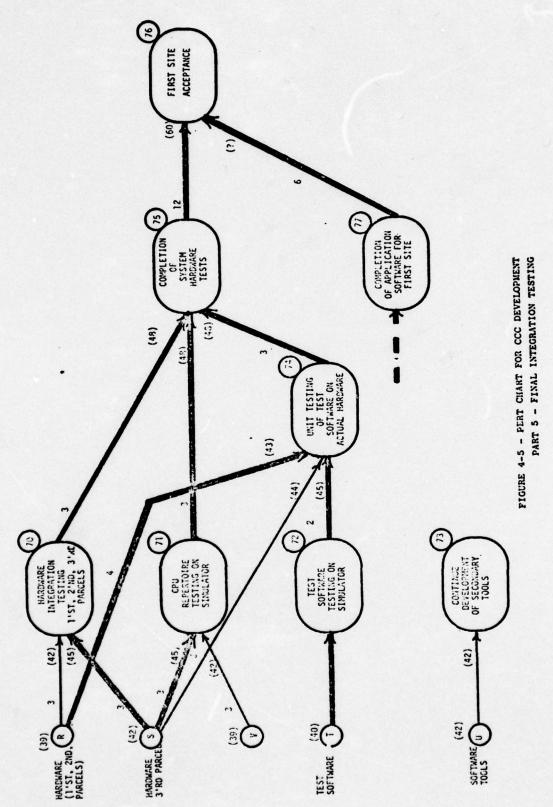
(1) Hardware design and testing

(2) Simulator testing of the various chips

(3) Hardware testing of the chips

(4) Design and encoding of the CPU micro-code

(5) Completion of final test software

(6) Simulator testing of the micro-code

(7) Verification testing of the system test software.

4.6 Phase 5 - Final Integration Testing

4.6.1 General

The previous phase started with a software and simulator dominance and ended with the beginning of hardware testing. This phase (refer to Figure 4-5) concludes with the final verification of working hardware.

FIGURE 4-4 - PERT CHART FOR CCC DEVELOPMENT
PART 4 - UNIT TESTING PHASE

FIGURE 4-5 - PERT CHART FOR CCC DEVELOPMENT
PART 5 - FINAL INTEGRATION TESTING

PREFACE

This volume, Volume II, contains a description of the Applications Baseline Study performed through a literature search.

From the Applications Baseline a Communications System Functional Baseline was derived which is described in paragraph 2 of the volume.

The last paragraph, paragraph 3, describes the evolutionary and creative processes that went into the derivation of the Central Computing Complex architecture.

These three paragraphs provide the reasons, rationale and justification for the CCC architecture.

These paragraphs were written by Mr. K. Hagstrom of North Electric Company and Dr. Boris Beizer of Data Systems Analysts, Inc.

VOLUME II

DEFINITION OF PROBLEM

TABLE OF CONTENTS

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Continued

LIST OF ABBREVIATIONS

AC	Alternating Current
ACF	Administration Control Function
ACK	An abbreviation for Acknowledge
AFTN	Aeronautical Fixed Telecommunications Network
AIOD	Automatic Identified Outward Dialing
AMME/ATTC	Automated Multi-Media Exchange portion of the Automated Telecommunications Center
AN/TTC-25	A tactical military circuit switching system
AN/TTC-39	A tactical military circuit and message switching system
ARPA	Advanced Research Projects Agency
ARPANET	(See ARPA)
ARTCC	Air Route Traffic Control Center
ART	Automatic Routine Tester
ARQ	Automatic Request-Repeat
ASCII	American Standard Code for Information Interchange. Also called USASCII.
ASF	Address Signaling Function
AUTODIN	Automatic Digital Information Network
AUTOVON	Automatic Voice Network
BASECOM	Base Communications
Baud	The unit of modulation rate. One baud corresponds to a rate of one unit interval per sec.
BH	Busy Hour
CAMA	Centralized Automatic Message Accounting
CBU	Conference Bridge Unit

CCC	Central Computing Complex
CCF	Compatibility Conversion Function
CCS	Common Channel Signaling
CCIS	Common Channel Inter-Office Signaling
CCITT	International Telegraph and Telephone Consultative Committee
CIF	Communications Interface Function
COMLOGNET	Communications Logistics Network
COMSEC	Communications Security
CP	Central Processor
CPF	Call Processing Function
CPS	Communications Processor System
CPU	Communications Processor Unit
C/S	Circuit Switch
CESE	Communications Equipment Support Element
CNCE	Communications Nodal Control Element
CSCE	Communications System Control Element
CSPE	Communications System Planning Element
C.T.N.E.	A packet switching network. Refer to Volume II, Section 1.
DAC	Direct Access Control
DC	Direct Current
DCS	Defense Communications System
DP	Dial Pulse
DID	Direct Inward Dialing

DPI	Data Processing Installations
DTMF	Dual Tone Multi-Frequency
E	Symbol for Erlang
E & M	A signaling arrangement characterized by the use of separate paths for signaling and the communication signals.
EMC	Electro-Magnetic Compatibility
EMI	Electro-Magnetic Interference
EOM	End of Message
ETS-4	Electronic Toll Tandem Switching System - 4-Wire
EVS	Electronic Voice Switching System
EWS-1	Electronische Wechsel Systeme (Electronic Switching System)
FAA	Federal Aviation Agency
FAA/AFTN	Refer to Volume II, Section 1.
FAA/WMSC	Refer to Volume II, Section 1.
GOS	Grade-of-Service
HAF	Header Analysis Function
H/R	Historical/Recovery
H/R/T	Historical/Recovery/Traffic
HSL	High Speed Line
ICCS	Integrated Communications Control System
ICMS	Integrated Circuit and Message Switch
IMP	Interface Message Processor
I/O	Input/Output
IS&R	Information Storage and Retrieval

ITA #2	International Teletypewriter Alphabet #2
ITS	Intransit Storage
IVSN	Initial Voice Switched Network
LAMA	Local Automatic Message Accounting
LDMS/ NAVCOMPARS	Refer to Section 2, paragraph 2.3.1.11.
LIF	Line Interface Function
LMF	Language Media Format
MCF	Maintenance Control Function
MF	Multi-Frequency
MPF	Message Processing Function
M/S	Message Switch
NACK	Negative Acknowledge
NCC	Network Control Center
NCF	Network Control Function
NCP	Network Control Point
NMF	Network Management Function
PABX	Private Automatic Branch Exchange
PCF	Path Control Function
PCM	Pulse Code Modulation
PMF	Position Management Function
PTC	Pentagon Telecommunications Center
PTT	Post Telegraph, Telephone
RAS	Random Access Storage
RMF	Remote Maintenance Function

LIST OF ABBREVIATIONS - Continued

R.I.	Routing Indicator
SAMSON	Strategic Automatic Message Switching Operational Network
SATIN IV	Refer to Volume II, Section 1
SENET-DAX	Slotted Envelope Network, Digital Access Exchange
SF	Single Frequency
S/F	Store and Forward
SITA	Society Internationale de Telecommunications Aeronautique
SSF	Supervision Signaling Function
SWIFT	Refer to Volume II, Section 1
TADSS	Tactical Automatic Digital Switching System
TARE	Telegraphic Automatic Relay Equipment
TCF	Transmission Control Function
TCCF	Tactical Communications Control Facility
TDM	Time Division Multiplex
TELEX/WUI	Refer to Volume II, Section 1
TENLEY	The name given to a family of communications security equipment
VDU	Visual Display Unit
VF	Voice Frequency
WMSC	Weather Message Switching Center

LIST OF FIGURES

1.0 COMMUNICATION SYSTEMS APPLICATION BASELINE

1.1 Purpose

The communication systems application studies were under-taken to establish an application baseline from which a functional baseline could be derived. Of the hundreds of processor controlled systems now in operation or in planning, only a small fraction could be studied in detail. The selection of systems to be studied, both data and circuit switching, was based on one or more of the following:

(1) The system was representative of a class of similar systems.

(2) It fit directly in with present and projected military communication requirements.

(3) It was, or is expected to be, successful and cost effective.

(4) It was, or will exist, in multiple copies.

(5) It contains some unique features which may have bearing on future system designs.

(6) It represents the current state-of-the-art within a specific switching mission category.

1.1.1 Some Definitions

One of the first problems to appear during the course of this study was one of semantics. Several distinct technical disciplines were involved, each with its own jargon, and it was always necessary to know to which discipline an individual belonged to understand exactly what he meant. For example, if a computer expert uses the word "register" he is usually referring to a logic device capable of storing digital information, but when a circuit switch designer uses the word, he is usually referring to a hardware device used to send or receive address signaling tones, pulses, or codes.

It is for this reason that the following abbreviated list of definitions is presented at this time. A more complete set of definitions of the terms and acronyms used in this report may be found in Volume VII, Glossary of Terms and Acronyms.

(1) Communications Processor System (CPS)

As used in this report, CPS refers to all operations
and logic processes necessary to perform the communi-
cation function. It includes all decision making
devices, i.e., all logic gates no matter where they
reside and all programs, whether wired-logic, firm-
ware or software.

(2) Communications Processor Unit (CPU)

As used in this report, CPU refers to that set of
processor hardware under the control of a distinct
Control Unit (CTU). It may include several Arithme-
tic Logic Units (ALU).

(3) Central Computing Complex (CCC)

As used in this report, CCC refers to a set of CPU's
complete with Input/Output (I/O) channels, processor
monitor units, channel units, control panels, some
form of data and program memory and a means to
interconnect them.

(4) Circuit Switch (C/S)

A complete system which possesses the capability to
establish real-time communication paths between
termination ports. During the "conversation phase"
or information transfer phase of the call, it is
completely transparent to the information being
transmitted.

(5) Message Switch (M/S)

A complete system which possesses the capability to
receive, store, and transmit digital information.
During the information transfer phase, it appears
as a variable delay line with or without multiple
taps.

(6) Blocking

This word takes on distinctly different meanings
when applied to message processing and matrix de-
sign. When used in the context of message processing
it means; "to organize a (binary) message into
modular sections or blocks consisting of a predefined
number of bits or characters".

When used in the context of matrix design it means; "a condition which exists in a matrix whereby a path between two matrix ports cannot be established due to existing conflicting paths, that is, a path cannot be established due to matrix congestion".

1.2 Circuit Switch Baseline Summary

The Circuit Switch baseline study was conducted to identify all switching system features required by systems now deployed and operational as well as systems being developed or planned. It was divided into three categories for research and evaluation purposes which are:
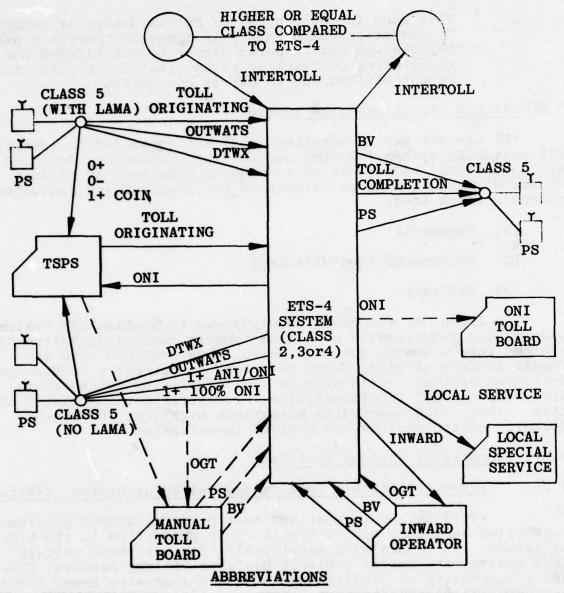
(1) Commercial

(2) Government (Non-Military)

(3) Military

The intent of the baseline study was to combine all system operational requirements of stored program controlled switching systems into a source document which, when combined with the identical type of study being performed concurrently on Message Switching systems, would provide the information necessary to arrive at the CPS functional baseline described in Section 2 of this volume. The succeeding paragraphs highlight the system features of the various C/S systems investigated.

1.2.1 Commercial Systems Analyzed

1.2.1.1 ETS-4 (Electronic Toll Tandem Switching System, 4-Wire)

ETS-4 is the largest and most powerful stored program controlled switching system presently in production in the U.S. or abroad. It is designed specifically for switching centers that control the nation's Direct Distance Dialing network, ETS-4 has a capability of handling up to 72,000 four-wire trunk terminations. This capacity, extensive translation, call routing, traffic and dial administration facilities and sophisticated office and network maintenance capabilities along with Network Control and Testing capabilities, makes the ETS-4 applicable in Class 4, 3 and 2 types of service. Refer to Figure 1.2-1. Stored Program Logic and data is used to direct all operations of the switching system. Centralized memory banks contain all information necessary for the operation of the system switching, maintenance and administration functions. Programs can be altered to accommodate changes in methods of operation. The system is modular and can be sized for either small or large offices to provide easy expansion to meet future traffic requirements.

TYPICAL NETWORK SURROUNDING AN ETS-4 SYSTEM

FIGURE 1.2-1

The switching equipment contains code switches arranged for four-wire switching. The switching matrix consists of trunk inlets and outlets, code switch matrix, and signaling devices. The four-stage, four-wire code switch matrix is configured in terminal groups of 800 trunk inlets (or outlets). A transfer subsystem acts as an adapter between the switching network and/or the input/output equipment and the processing system.

The common control subsystem contains four basic elements:

(1) Processor groups with 1 to 7 dual processor units.

(2) Data stores in which all data (permanent, as well as variable) is stored.

(3) Transfer control units acting as intermediaries for all communication between the data processing system and the switching equipment.

(4) Multiplexer groups which provide the processors with access to the transfer control and the data stores.

The common control subsystem is a modular, real-time, multiprocessing configuration. Each processor of the group has associated with it, its own data store, transfer control, and multiplexer.

The common control system functions are implemented with duplicated real-time software programs. The software for the system consists of approximately 300,000 instructions and typically has 1 million words of data associated with an application. All control programs operate under a real-time Executive program. The program hierarchy and data structures maximize processing capacity and real-time response. Extensive provisions are made for automatic system re-configuration under software control.

The software system for network management provides the means for controlling traffic which substantially exceeds the capacity of the Direct Distance Dialing network, or a unit of it, thereby reducing or avoiding serious degradation of service.

Self-checking and self-routing is provided by the software system for all hardware subsystems. The software also performs diagnostic procedures during low traffic periods when it is not engaged in call handling. Upon determining trouble, the system diagnostics feature provides a direct readout advising of the fault.

The maintenance center is a control console which allows

maintenance personnel to communicate with the common control subsystems under command-and-control software. The maintenance center features provide routine testing, fault location, problem identification, and status control of the various units.

1.2.1.2 NX-1E

The NX-1E is an electronic common controlled stored program switching system designed for Class 4 and 5 Central Office applications, manufactured by North Electric Company. Utilizing a crossbar switching matrix, the system is capable of providing 32,000 subscriber lines and 64,000 directory numbers*. The processing complex is comprised of pairs of identical APZ-142 processors. Within a pair of processors, one is always "on-line" and one is held on "standby" ready to automatically assume the work load should the "on-line" processor fail. A specific system may be equipped with up to eight (8) processor pairs depending upon the system size of the office being equipped. Subscriber features include Call Forwarding, Recorded Message, Call Transfer, etc., which are provided with most stored program systems presently available.
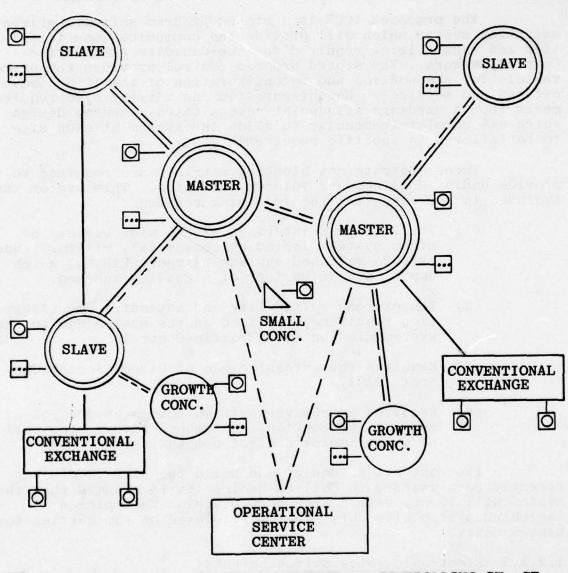
Optional system features available include Local Automatic Message Accounting (LAMA) and Centralized Automatic Message Accounting (CAMA).

* Provision for party-line services.

1.2.1.3 EWS-1

The EWS-1 (Electronic Switching System) is a stored program controlled telephone exchange manufactured by Siemens Corporation, capable of functioning as a Circuit Switching system handling from 100 to 20,000 subscriber lines. Utilizing a glass Reed Space Division Matrix, the EWS-1 provides a wide variety of subscriber features (announcement service, abbreviated dialing, automatic wake-up service, etc.), under software control. Its (EWS-1) inclusion in the baseline study was not due to any unique Circuit Switch functions provided, since the subscriber features are nearly identical to other similar stored program systems, but because the proposed EWS-1 network environment does employ a distinctive control feature. (Refer to Figure 1.2-2). The use of a central computer facility (Service Center) connected via a common channel signaling link to all "Master" exchanges located throughout the network, provides the capability of relieving the "Master" processors from the requirement of storing data and programs which are not frequently used. Thru-connect of the common channel signaling link permits the Service Center access to the "Slave" EWS-1 switching system thru the "Master" switching matrix.

This feature, i.e., remote machine-to-machine communication via switchable links included in a Circuit Switching system, bears directly on the problems addressed by this study.

II-6

LEGEND:

⬚ DATA TERMINAL   MASTER = CONTROLLING SW. CT.

◎ SUBSCRIBER   SLAVE = CONTROLLED SW. CT.

—— SPEECH PATH

--- SIGNALING CHANNEL

EWS NETWORK CONFIGURATION

FIGURE 1.2-2

1.2.2 Government Non-Military Systems Analyzed

1.2.2.1 Integrated Communications Control System (ICCS)

The proposed ICCS is a stored program software driven switching system which will provide the communications facilities and capabilities required for the Canadian Air Traffic Control Network. The stored program control provides the primary vehicle for the control and reconfiguration of the ICCS communications and services. Reconfiguration and varying site requirements from a hardware standpoint necessitated a system design which was completely modular to allow the system at each site to be tailored to specific requirements.

Three separate non-blocking matrices are required to provide Radio, Hotline, and Telecom switching. This system was included in the study for the following reasons:

  (1) The ICCS must interface with a wide variety of other systems including commercial, military, and private switched systems (Airport PABX's) which employ a wide variety of signaling schemes.

  (2) The extreme reliability and automatic reconfiguration requirements placed on the system due to the system mission (Air Traffic Control).

  (3) Requires the extensive use of Direct Access Control (DAC).

  (4) Requires extensive position equipment control of readout information, equipment status, etc., all of which impact control complexity.

The contract to design and build the nine ICCS's has recently been awarded to Collins Radio. It is assumed that the system will incorporate a Frequency Division Multiplexed switching system with data links multiplexed on the carrier for system control.

1.2.2.2 Electronic Voice Switching (EVS)

The EVS system would have provided voice communication required to link the twenty-two geographical Air Route Traffic Control Centers (ARTCC) located throughout the United States as well as the voice communication required within the ARTCC. The system must interface with a wide variety of Military, Commercial, and private communications systems including the ICCS previously described. Similar in function to the ICCS, the reconfiguration and varying site requirements necessitates a system design which is completely modular to allow the systems at each

site to be tailored to the specific site requirements.

The system was included in the study for the following reasons:

    (1)   Requires extensive use of DAC, up to thirty DAC buttons per position, which required data links from each position directly to the central processor system.

    (2)   Employed remote control of Remote Control Air Ground (RCAG) radio sites with local switching at the site, under EVS control, via a control channel.

    (3)   Required a broadband data link switching matrix in addition to the others.

Originally awarded to Philco-Ford, the contract was cancelled during the course of this study. A revised specification is anticipated to be submitted for response during 1976.
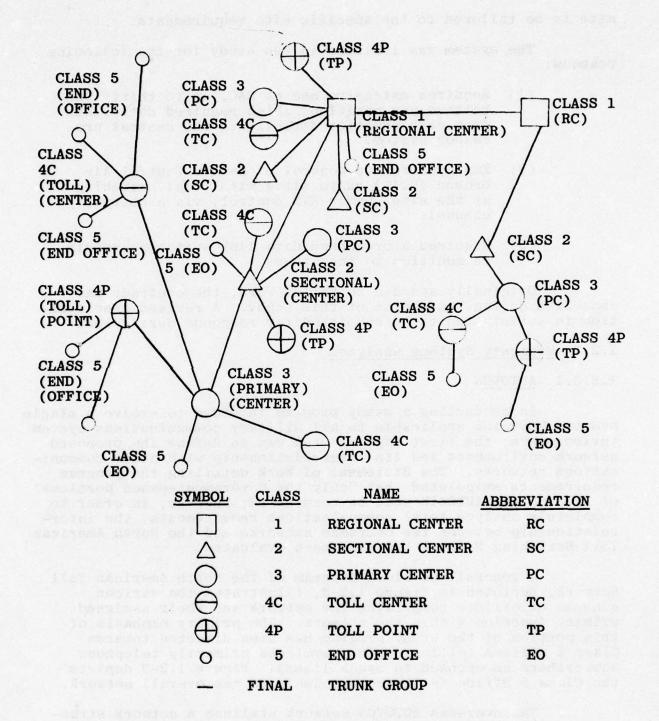
1.2.3 Military Systems Analyzed

1.2.3.1 AUTOVON

In conducting a study program intended to evolve a single systems approach applicable to all Military communications system environments, the first logical step was to define the proposed network environment and its interrelationship with other communications networks. The Statement of Work detailing the program requirements stipulated that "only the Government-owned portions" of AUTOVON and AUTODIN need be considered; however, in order to completely analyze total communications requirements, the interrelationship between the overseas networks and the North American Toll Switching Network has also been evaluated.

A generalized block diagram of the North American Toll Network, depicted in Figure 1.2-3, illustrates the various classes of offices comprising the network and their assigned primary function within the network. The primary emphasis of this portion of the study program has been directed towards Class 5 offices (offices which terminate primarily telephone subscribers as opposed to trunk lines). Figure 1.2-3 depicts the Class 5 office in its relationship to the overall network.

The overseas AUTOVON network utilizes a network structure somewhat different than the "Hierarchical" structure depicted in Figure 1.2-3. Since survivability is an additional network requirement not placed on commercial networks (where economics dictate the network structure), the AUTOVON system

CLASS 4P (TP)

CLASS 5 (END) (OFFICE)

CLASS 3 (PC)

CLASS 4C (TC)

CLASS 1 (REGIONAL CENTER)

CLASS 1 (RC)

CLASS 4C (TOLL) (CENTER)

CLASS 2 (SC)

CLASS 5 (END OFFICE)

CLASS 2 (SC)

CLASS 5 (END OFFICE)

CLASS 5 (EO)

CLASS 4C (TC)

CLASS 2 (SC)

CLASS 3 (PC)

CLASS 3 (PC)

CLASS 4C (TC)

CLASS 4P (TOLL) (POINT)

CLASS 2 (SECTIONAL) (CENTER)

CLASS 4P (TP)

CLASS 5 (END) (OFFICE)

CLASS 4P (TP)

CLASS 5 (EO)

CLASS 3 (PRIMARY) (CENTER)

CLASS 5 (EO)

CLASS 4C (TC)

CLASS 5 (EO)

SYMBOL	CLASS	NAME	ABBREVIATION
☐	1	REGIONAL CENTER	RC
△	2	SECTIONAL CENTER	SC
◯	3	PRIMARY CENTER	PC
⊖	4C	TOLL CENTER	TC
⊕	4P	TOLL POINT	TP
◦	5	END OFFICE	EO
—	FINAL	TRUNK GROUP	

NETWORK DIAGRAM
NORTH AMERICAN TOLL NETWORK

FIGURE 1.2-3

employs a "polygrid" network structure (Figure 1.2-4), which provides the survivability capability necessary in Military communications. Each office in the "polygrid" network provides services which can be compared to a combination Class 4 and Class 5 office in the North American Toll Network.

Since the AUTOVON network is in a constant state of evolution, the features and services available at each switching center location vary considerably. The CONUS AUTOVON switching centers do provide a wide range of subscriber features (conferencing, compressed dial, etc.), found in commercially equivalent state-of-the-art switching centers.
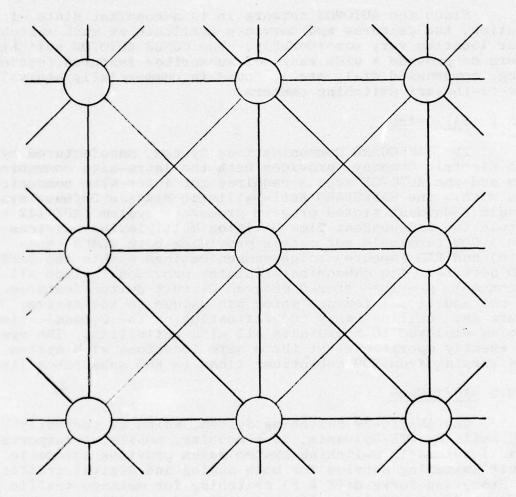
1.2.3.2 SAFEGUARD

The SAFEGUARD Communications System, manufactured by North Electric Company, provides both the intra-site communications and the AUTOVON access required for inter-site communications within the SAFEGUARD Anti-Ballistic Missile Defense system. A single redundant stored program processor system (APZ-142 type) controls two independent Time Division Multiplexing matrices of up to 1,008 terminals per matrix providing both BLACK (non-secure) and RED (secure) voice communications within the SAFE-GUARD network. The subscriber features provided include all of the commonly provided stored program Circuit Switch features with one additional feature which was unique to the system. The feature is: calling party identification on the Communications Consoles employed to coordinate all site activities. The system is presently operational at three site locations with system sizes varying from 400 subscriber lines to 800 subscriber lines.

1.2.3.3 AN/TTC-39

The AN/TTC-39 Switching System, which is currently being built by GTE/Sylvania, is a modular, mobile, transportable tactical automatic switching system which provides automatic circuit switching service for both analog and digital traffic, with Store-and-Forward (S & F) switching for message traffic provided by add-on modules. It is capable of providing concurrent circuit and message switching, or either of the two independently, as required by the user. Its design incorporates the use of micro-electronics and state-of-the-art design concepts to minimize size, weight, and power consumption while providing all of the services and features required of a modern automatic tactical switching system.

The equipments, which together comprise an AN/TTC-39 system, are designed to allow their installation in shelters, for use in a mobile and transportable field environment, in vans, for use as semipermanent transportable installations, or in permanent buildings for fixed plant installation use. In

AUTOVON NETWORK GRID

FIGURE 1.2-4

addition, they are designed to be modular to permit each system, whether shelter, van, or fixed plant mounted, to be configured to meet the user's requirements efficiently and effectively. The AN/TTC-39 system provides the facilities to allow gradual, orderly, controlled, and efficient conversion from an essentially analog communication network to a secure digital communication network, maintaining node and network compatibility with both old and new switching systems and end instruments.

The "standard" features and operational capabilities of the AN/TTC-39 include all of the features and capabilities of the most recently developed automatic tactical switching systems which include, among others, the AN/TTC-25, -30, -37, and -38 systems.

The AN/TTC-39 system also incorporates several features and operational capabilities which are for the first time combined in a single tactical switching system. The most significant among these features and capabilities are:

- Common channel out-of-band trunk signaling

- Secure and non-secure traffic handling in the same switching matrices

- Analog and digital traffic handling in the same switching system

- Message switching and circuit switching combined in the same network node

- Circuit switch and circuit switch technical control combined as one integrated design

- Wideband analog signal switching

- Compliance with EMC/TEMPEST requirements

- Interface with existing inventory COMSEC equipment

- AUTOVON and foreign system interface

- Capability to be converted from a 100% analog node to a 100% digital node in modular steps

1.2.3.4 BASECOM

Any comprehensive study of Military communications requirements must give significant emphasis to the specific demands of communications requirements of Military bases as a separate entity (divorced from network requirements). These requirements

are referred to under the general heading of "BASECOM". The term BASECOM includes the following communications functions applicable to this study:

    (1) Base Telephone, inside plant and systems.

    (2) Base record communications and information transfer systems.

    (3) Communications support for data processing systems.

    (4) Telecommunications interface with off-base communications systems.

The Mission Analysis on Air Force Base Communications-1985, dated April 1973, detailed the Circuit Switching functions required on Air Force installations, and was used as a general guide in determining circuit switching functions and features that must be considered in arriving at a CPS architecture.

1.2.3.5 AN/TTC-25

The AN/TTC-25 (or AN/TTC-37) is a stored program controlled tactical electronic switching center manufactured by North Electric Company for use in tactical military environments. Employing a four-wire time division matrix varying in size from 344 line/trunk terminations (AN/TTC-25) to 620 line/trunk terminations (AN/TTC-37), the matrix is controlled by a marker logic which is firmware implemented, while the remaining call processing logic is software controlled. The central processing subsystem consists of dual APZ-142 processors, each having access to 20,000 words of memory.

Traffic handling capability of approximately 22 CCS (.61E) per terminal is provided due to the anticipated high busy hour peak traffic loading (alert conditions, etc.). Subscriber features provided include conferencing, line grouping (transfer or busy), five levels of subscriber priority assignment, operator recall, etc.

1.2.3.6 NICS IVSN

NICS IVSN stands for NATO Integrated Communications System (NICS) Initial Voice Switched Network (IVSN). The IVSN will consist of an initial network of 24 switching systems with at least one switch in each of the NATO member countries. The procurement cycle, i.e., the internation invitation for bids, began in 1975 with contract award being anticipated in the first quarter of 1976.

The switching systems must be processor controlled and modularly expandable from a minimum size of 250 terminations to a maximum of approximately 750 terminations. The important features of the IVSN Access Switches and network are:

(1) Serves both directly connected subscribers and subscribers connected through various PABX's and PBX's.

(2) Tandem switching.

(3) Common Channel Signaling tailored after CCITT #6 signaling schemes.

(4) Wideband switching.

(5) Fixed plant construction.

(6) Future automatic Network Control exercised from one or more points within the network.

(7) Expansion of the network to more than 50 nodes.

(8) Future incorporation into the network of a Nodal network with the current IVSN switches acting as Access Switches.

(9) Incorporation of inventory crypto equipments as an option.

(10) Complete Traffic metering reported automatically or on request to either the local maintenance position and/or the Network Control Center via the Common Channel.

(11) Remote maintenances and diagnostic reporting via the common channel.

The main stress of the IVSN is, as expected, survivability and reliability. This network is the first military network to make extensive use of common channel signaling and to use the common channel for network control and remote maintenance.

1.3 Message Switch Baseline Summary

The following list contains those systems and partial systems considered.

Military

AMME/ATTC - Automated Multi-Media Exchange/Automated Telecommunications Center

AUTODIN   - Automatic Digital Information Network

PTC   - Pentagon Telecommunication Center

AN/TTC-39 TRI-TAC

Military (Canadian)

SAMSON   - Strategic Automatic Message Switching Operational
             Network

Government (Non-Military)

ARPA   - Advanced Research Projects Agency

TARE   - Telegraphic Automatic Relay Equipment

Commercial

SITA   - Society International de Telecommunications
             Aeronautique

SWIFT   - Society for World Wide Intrabank Financial Telecommun-
             ications

TELEX/WUI - TELEX/Western Union International

Partial Systems (Types)

Base Distribution

Pentagon AMPS-P

Fort Richie AMPS-R

Navy LDMS/NAVCOMPARS

General Store and Forward Switching Systems

TADSS - Tactical Automatic Digital Switching System

FAA/WMSC - Federal Aviation Administration Weather Message
          Switching System

FAA/AFTN - FAA Aeronautical Fixed Telecommunications Network

497L System

SATIN IV - Strategic Total Integrated Network

Western Union International Leased Channel System

NABPARS

British Post Office

Packet Switching Systems

C.T.N.E.

British Post Office

Special

ICMS - Integrated Circuit/Message Switch

1.3.1 Systems Analyzed

1.3.1.1 AMME/ATTC - Army

The Automated Multi-Media Exchange (AMME) portion of the Automated Telecommunications Center (ATTC) represents the applications of message and data switching oriented towards military communication centers to meet distribution requirements, staff communications automation and high speed interface with high volume subscribers via a variety of transmission media. The system's design concept resulting in functional specifications, issued in January of 1972, were several years in development and composition. The AMME/ATTC system is currently under development by Univac under contract to the General Services Administration (GSA). The system switches have been implemented by Univac using Univac 418-III. In general, the AMME/ATTC system is representative of functional requirements related to base distribution of local message and data traffic as well as performing the function of a gateway to long haul communication networks such as AUTODIN. The functional requirements of the AMME/ATTC system are significant guidelines for the functional roles of the CPS processor system.

The Automated Multi-Media Exchange (AMME) portion of the ATTC system is a full function store and forward message switching system which provides, in addition, some circuit switching facilities. Together these are intended to fulfill the following roles:

    (1) A local (base) message and data switching system to satisfy local, on-base communications requirements.

    (2) To provide communication interface with base data processing installations to allow connection of terminals and networks of terminals to such DPI installations.

(3) To serve as a gateway for traffic routed over the DCS long haul network and to serve as a local distribution center for traffic received from the AUTODIN system.

Within the AMME switch system, both data and narrative traffic is fully accounted for and automatically switched while maintaining full compliance with COMSEC and requirements as well as priority and precedence distribution.

Distribution is by plain addressing for local messages, including official title, message content and message destination. The AMME design specification called for the provision of circuit switching connections between compatible terminals to deviate processing burden for traffic between such terminals. In this mode the system (switch) is to be transparent to circuit switch traffic, and the connection for circuit switching would be done in two stages. The terminal requesting a circuit switch connection would initiate the request addressing itself to the data switched AMME system. The latter attempts to connect to the calling party, and after establishing connection will effectively remove itself from the interface until the receipt of an "off-hook" signal from either party for other kinds of intervension/preemption.
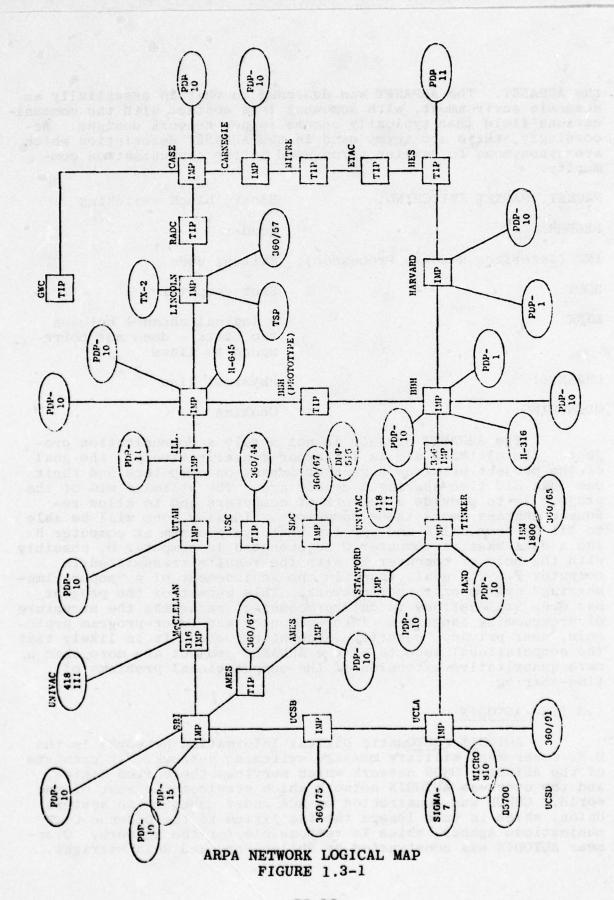
1.3.1.2 ARPA

The ARPA (Advanced Research Projects Agency) network (Figure 1.3-1) is a high level communications network. The network was initiated in early 1969, some months after the SITA high level network went into commercial operation. However, it appears that the ARPA network design was done independently of the earlier SITA network.

The ARPANET has been growing in size and function since its inception. It now services more than 30 terminals at a number of widely separated locations. Additional, experimental, links have been established to Hawaii and England. The network is in a continuing state of flux with regards to services offered, routing procedures, protocols, programs, etc., as befits its experimental nature.

Almost all the changes that have occurred in the network design, have been towards increasing the services offered, with the expected resulting decrease in throughput. Part of the ongoing research effort has been improvements in the routing scheme which, while decreasing network congestion and increasing throughput on a network level, will result in an inevitable decrease in individual switch throughput.

Some notes on terminology are in order when discussing

ARPA NETWORK LOGICAL MAP
FIGURE 1.3-1

II-19

the ARPANET. The ARPANET was designed in what is essentially an academic environment, with somewhat less contact with the communications field than typically occurs in new network designs. Accordingly, there are terms used in the ARPANET description which are synonymous to earlier terms used in the communication community.

PACKET, PACKET SWITCHING:	Block, block switching
LEADER:	Header
IMP (Interface Message Processor):	Switch, node
HOST:	User computer
LINK:	A logical channel between two hosts - does not correspond to lines
CHANNEL:	Physical line
QUENCHING:	Choking

The ARPANET project is not merely a communication project. Its ultimate aim is far broader qualitatively: the goal of the project promises a greater impact on computers and their use than did time-sharing a decade ago. The utlimate aim of the project is to provide a network of computers and to allow resource sharing among these computers. Ideally, one will be able to sit at terminal A and specify that: a program at computer B and a data base at computer C be executed in computer D, possibly with the help of computer E, with the results transmitted to computer F. The goal, then, is the achievement of a "super-time-sharing" system or rather network. This aspect of the project has deep implications in data processing, regarding the structure of programming languages, inter-computer and inter-program protocols, user privacy, security, accounting, etc. It is likely that the computational aspects of the ARPANET project are more than a mere quantitative extension of the computational problems of time-sharing.

1.3.1.3 AUTODIN

AUTODIN (AUTOmatic Digital Information Network) is the U.S. Government military message switching network. It consists of the AUTODIN CONUS network which services the United States, and the overseas AUTODIN network which services the rest of the world. CONUS was constructed by RCA under contract to Western Union, which in turn leases the facilities to the Defense Communications Agency, which is responsible for the network. Overseas AUTODIN was constructed by Philco-Ford and sold outright.

While there are hardware and software differences between the CONUS and overseas segments, operationally, they are part of the same network and these centers behave in much the same way.

AUTODIN was originally conceived as the Air Force COMLOGNET (COMmunication LOGistics NETwork). Responsibility was shifted from the Air Force to DCA and its role expanded to handle logistics traffic for all services, as well as paramilitary and other branches of the executive. It serves as a trunking network for some 13 other, smaller Government networks. As such, AUTODIN inherited a variety of terminals, formats, line speeds, and procedures, which could not be completely revamped without an excessive loss of capital investment.

A fact often disregarded in the evaluation of the AUTODIN network is that it is a military network. Reliability is stressed more so than in most other networks. Trunks and trunking plans are based first on network survival and peak load requirements that might occur during an international emergency, and only secondarily on the basis of average traffic requirements. As any military system it is grossly over-designed for operation during average conditions, and can be inadequate in emergencies.

1.3.1.4 Pentagon Telecommunication Center (PTC)

This center is presently in the planning stage. Detailed specifications have not yet been developed. However, the concept is sufficiently matured to have warranted an investigation of its impact vis-a-vis the CPS. When completed, it will probably be the extreme end of the base distribution system concept.

The PTC is intended to integrate the operation of several communication systems now in the Pentagon. These include: AMPS-P, AMPS-R, PTC (Army), CNO-TCC (Navy), CMC-TCC (Marine Corp/Bupers), AF-TCC (Air Force). Operationally, each of the component users of the system are to appear to have their own dedicated system. Minimal operational compromises will be made.

USAF-TCC

This function is presently serviced by dual UNIVAC DCT 9000/2000 processors for narrative traffic and UNIVAC SPECTRA 70/45 for data traffic. Additional, near term plans include the implementation of an LDMX and the replacement of the UNIVAC DCT 9000 with UNIVAC 418-III.

USASTRATCOM-PTC

Narrative traffic is handled by dual RCA 3301 systems and data traffic handled by dual IBM 360-50 computers. The PTC

contains a 30 day reference storage. It is augmented by additional subsystems for remote message entry for narrative traffic. Multiple VDU's are used for editing and retrieval.

JCS-AMPS

There are two AMPS sites: AMPS-P at the Pentagon and AMPS-R at Fort Richie. Both sites are Burroughs D-825 dual computers. OCR equipment provides message entry. Multiple VDU's are used for message entry, edit and retrieval.

USN CNO-TCC

This system consists of an LDMX (UNIVAC SPECTRA 70/45 with UNIVAC 70/1600 as front-end). It contains a large on-line data base, has heavy retrieval requirements, VDU's for edit, etc.

USN (NAVPERS-TCC)

This function is presently handled by a MODE V AUTODIN terminal augmented by several data terminals.

U.S. MARINE CORPS (MCM-TCC)

This function is presently supported by an IBM 360/20 AUTODIN terminal and is otherwise not automated.

1.3.1.5 SAMSON Message Switching System (CND)

SAMSON (Strategic Automatic Message Switching Operational Network) is to become the Canadian Government's military message switching network designed to satisfy the needs of the Department of National Defense in the 1975-1985 time-frame. The network and system functional concepts for the SAMSON system have been in development for the last three to five years. Inputs into this effort have been generated from many sources, including official Government technical policy decisions as well as significant study and analysis undertaken by private contractors under the direction of the Canadian Department of National Defense. In short, exhaustive investigation of existing networks were made to establish and define the most desirable features for long haul inclusion into the SAMSON network. Many studies were conducted, including: adaptive routing, accountability concepts, network survivability concepts, as well as other features not presently employed in message switching systems.

The total SAMSON message and data switching system is a sophisticated, five node store-and-forward communications complex for the management and transmission of strategic and tactical communications of the Canadian Department of National Defense.

Three of the five nodes are totally connected and comprise the SAMSON backbone system. The remaining message switching centers include the maritime semiautomatic exchanges and the supplementary communication exchange. These exchanges are designed to support the needs of maritime and military aircraft communications while the supplementary communication exchange is designed to support special military community communication requirements.

The SAMSON system is designed to pass traffic originated in several different codes and formats and at widely varying speeds. The SAMSON system is designed to provide rapid, secure, and totally accountable message processing service to local users connected either directly to a switching center or through remote concentrators to the center, as well as providing longhaul message trunking for transcontinental (as well as transoceanic) communications.

The existing military communications systems and equipment - primarily composed of 50 to 75 baud TTY's are to be totally absorbed into the new system which should continue to supply and support the needs of these users without interruption and with a minimum imposition of new procedures and techniques.

The SAMSON system is wholly designed as a military system supporting strategic and tactical communications needs. Therefore the objectives of reliability, viability, survivability, and security as well as total node-to-node message accountability, and integrity were placed at a premium level of importance in terms of design considerations. In this manner the SAMSON system is similar to the mission of the DCS Network. The required throughput and network topology are designed to meet the needs of network survival and peak load requirements that might occur during an emergency and as such may seem to be somewhat overstated for average traffic conditions projected for the next decade.

1.3.1.6 SITA

The SITA (Society International de Telecommunications Aeronautique) Network is an international network with centers in Amsterdam, New York, London, Paris, Brussels, Frankfurt, Rome and Madrid, and Hong Kong. In addition to these major centers SITA maintains minor switching complexes, concentrators and other facilities at over 150 locations throughout the world. Philips of Netherlands constructed the London, Paris and Amsterdam sites while UNIVAC constructed the remaining major switching centers. The functional requirements for all the centers are essentially identical, however, there are some significant differences in the details of the hardware and software implementations between the two vendors.

The SITA system represents one of the most sophisticated and successful applications of packet switching techniques available to date. The SITA computer complex includes nine computerized switching centers linked by 2400* BPS synchronous full-duplex leased lines. Topologically the high-level network includes the connectivity of each center to at least two others and up to four high-level trunks for the Paris, Frankfurt and London centers.

The equipment configurations at the nine centers include dual UNIVAC 418-MOD-III's for the New York, Brussels, Hong Kong, and Frankfurt, Madrid and Rome centers. Each of the UNIVAC 418's have 65K of core with a 2 microsecond cycle time. The remaining centers - Amsterdam, London, and Paris are configured with the Philips' DS714-MARK II communications switching computer with 256K core and 2 microsecond cycle time.

The total terminality for the SITA system is currently populated at about 7,000 terminals. While most of these are 75 baud teletypes there is also a number of directly connected computers for administrative message processing and reservation availability and accounting procedure. Some standard interfaces have been developed for agent set devices. These include the UNIVAC uniscope 100, RAYTHEON's PTS-100 and IBM's 2915. Similarly, there are interfaces with the different computers being used for individual airlines reservation systems.

Most teleprinter terminals are directly linked to subsidiary centers, e.g., in Lisbon, Luxembourg, Prague, Vienna, Zurich, Athens, Katmandu and Hong Kong. The subsidiary centers collect traffic and transmit it to the high-level network. While some of these are still torn tape operations approximately twenty are computerized using Thompson-Houston 4020, Tatheon 766 or 704 satellite processors. They are connected to the HLN via 2400 BPS circuits. Other centers are equipped with UNIVAC multiplexers. There are still 125 manually operating reperforating centers which communicate to the HLN via 50 baud circuits.

1.3.1.7 SWIFT - Commercial Processing System

The SWIFT system (Society for World-Wide Intrabank Financial Telecommunications) is a system designed to be used in a new international telecommunications network among some 250 member banks. The transactions among these banks include transfers of payments and other message traffic associated with international banking. The member banks are spread out in some fifteen countries with initially one in Canada and the United

---

* Being increased to 4800 baud in some cases.

States and the rest in European countries. The SWIFT system is currently under development and is scheduled for initial implementation during the early part of 1976.

At the time of this writing the Burroughs Corporation was awarded the contract to implement the SWIFT system. They plan to use the Burrough's B3700/4700 computers as the host processors for the switching centers coupled with the Burroughs B774 front-end communications processor. A series of remote cascaded concentrators, the B775 are used to directly connect to the remote terminals.

The SWIFT system provides communications services to over 250 member banks in fourteen European countries and Canada and the United States. Initially the switching system is to provide a vehicle for the communication among the member banks for administrative and other banking transactions. The system is not intended to be a data processing clearing house for accounting balancing but truly a message switching service in which the message text is transparent to the system. Ultimately, it is planned that some measures of data processing functions would be augmented into the switching system to provide for the beginnings of possible clearing house activities or otherwise balancing and reconciliation type financial functions.

Initially, the system is designed and implemented with the ultimate line termination and terminal connectivity unspecified. In any event, the message switching centers are essentially buffered from all line termination, protocol and procedure problems through the use of remote concentrators which essentially act as buffering and packaging stations for message input and message delivery. While traditional military-oriented requirements of extreme security are not imposed on the SWIFT system a full accountability procedure as well as a multi-level precedence scheme and a full set of ancillary message switching functions are included in the systems functional concept.

1.3.1.8 TARE

"TARE" stands for Telegraphic Automatic Relay Equipment. This network is intended to handle the message and circuit switching requirements of the NATO community. Nineteen sites are planned to be deployed throughout Europe over a period of time extending to 1980. The network is presently in the planning stage and final specifications have not yet been released.

Line speeds range from 50 to 2,400 baud with the preponderance of lines in the low speed region. A typical switch will have from 50 to 150 lines. Communications includes satellite links. Full-duplex, half-duplex and simplex lines are to be used. The trunk line procedure is unusual in that its ACK's and

NACK's are returned on an associated dedicated "order-wire" at a low speed. The variety of terminals and line procedures is impressive. In addition to TARE interfaces, the switch must communicate with a large number of pre-existing terminal devices, none of which are likely to be compatible. Furthermore, there will be interfaces to a variety of national and international military networks such as AUTODIN. Probably the most trouble-some interface will be with the local PTT (Post-Telegraph-Telephone).

1.3.1.9 TELEX/Western Union International (WUI)

The Western Union International TELEX switch is a fully automated switching complex presently under development for the Western Union International. The system is designed to replace an existing manual and semi-automatic switching complex and will provide not only significantly faster and more economical service to TELEX users, but also provide them with greatly increased ser-vice support in terms of available functions and message handling assistance. It is scheduled for cutover within the current calendar year.

The system is designed to respond to the rapidly in-creasing demand of existing users and the increasing number of users themselves. The TELEX message switching center is designed to be fully automated to provide TELEX - as well as TWX Bell System teletypewriter exchange users with rapid, efficient, and more functional data switching services.

While the switching complex is not a circuit switch per se, it provides to the existing community of some 7000 to 10,000 domestic and international users the appearance of direct point-to-point real-time message communications facilities. In addition the system provides fully automated message store and forward service as a user selected option.

While many of the lines are directly connected TELEX subscribers, several hundred of these lines are trunks to other switching centers including the Bell System TWX Exchange as well as overseas TELEX exchanges in London and Paris for example. The switching facilities fully support both the standard five level, 50 baud TELEX circuits and teletypewriter equipment, as well as the eight level ASCII, 50, and 100 baud TWX circuits and termi-nals. This system is not designed to accommodate high-speed voluminous data transmission traffic.

1.3.1.10 AN/TTC-39 TRI-TAC Store and Forward Switch

The AN/TTC-39 system concepts were developed to provide the communications backbone to the various tactical programs of the several services. The current TRI-TAC program has just com-pleted its first phase of development in which two prototype

models of the switching complexes were constructed by teams of American companies working independently. In addition to the prototype model, complete detailed engineering development designs were produced to guide the phase two effort of full model production and initial deployment in a limited number of actual sites. The recent award of the Phase Two effort enables us to include a discussion of this important message switching system in this baseline study without endangering any privileged information on the part of one or more vendors.

An example of the TRI-TAC is in its role as the communication element of the ARTADS program. The ARTADS program requires a high degree of mobile unit communications both vertically - from unit command(ers) up through the command hierarchies to the Army levels. In addition, horizontal communications are also required among and between units at the same level - this applies to all levels in the military operational command structure. Furthermore, these communication channels are required precisely in theaters of operations where there are no existing (friendly) communications media. Thus, the communications must go where the action is, and communications must be maintained at all times. This, in essence, is the TRI-TAC mission. It requires the deployment of communications equipment in highly mobile, hostile, and dynamically changing configuration environments in which yesterday's switch may be gone today, but communications must be maintained.

The phase one effort of the TRI-TAC program was designed to isolate the areas of risk (deployment of technical resources) and reduce them to the greatest extent possible, by selecting tactically qualified communications processing equipment, and combine this with field-proven store-and-forward message switching systems designs and concepts. The initiation of the phase two part of the program suggests that these goals have been met by the engineering design plans developed in phase one.

Among the many functional requirements of the TRI-TAC message switching system deployment several have significant impact on the operations, functions, and performance of the equipment and related software modules:

    (1)   The utilization of the same host computer components for both the circuit and message switching components - although these are separately deployable.

    (2)   Complete compliance with militarized transportability, and housing in a prescribed S-280 shelter(s).

(3) Full-fledged store and forward accountable message switching functions in a highly dynamic and mobile connection and routing environment.

(4) Specific facility to be adaptable to rapidly changing network topology in terms of both trunks to other switches, and connections to directly connected subscribers.

(5) The ability to take over full operations of other switches which are temporarily (or permanently) out of service.

(6) Capability of being quickly placed in service with known and unknown routed subscribers so as to achieve as fully automatic S&F operations possible in a tactical deployment.

These requirements and others (particularly in the area of security and EMP protection) placed unique design requirements on the applicable CPU equipments and the architecture and sizing of the software modules. Among these are included the following:

(1) Supports directly connected subscribers with terminals of dissimilar codes, modes, speeds, and message formats.

(2) Provides complete journal and reference storage for all messages switched through the S/F module.

(3) Provides message retrieval.

(4) Automatic multiple routing of messages.

(5) Message and communication line security provided through software message monitoring; and COMSEC equipment.

(6) The communications components are modularly expandable in groups of lines - 25, 50, 100, 150.

While no one of these message switching functions is out of proportion for a large scale fixed based message system, their incorporation into a small-to-medium scale tactical switching system placed somewhat unique demands on the integration of total computer system resources and processing strategies. In particular, network recoverability and reconfiguration placed stringent demands on provisions for routing management and deeper historical storage of messages for retrieval.

Stringent security requirements are imposed throughout

the entire TRI-TAC system configuration and field network deployment. TRI-TAC is designed to support the communications requirements of several special interest communities serving military and defense operations. Seven degrees of security classification and attendant processing is required and specified. In addition, the security categories are open-ended so that additional categories can be added as needed as field modifications to the operational system.

1.3.1.11 Other Systems

Of the many systems studied, several deserve special mention rather than a complete review as given in the previous section, and in lieu of being ignored completely.

The systems outlined below were studied and compared to those reviewed above, with the distinguishing criteria.

(1) Was the salient functional and performance features (characteristics) significantly different from those selected for review?

(2) Did the system add any new input to a baseline of functional requirements pertinent to the CPS architectural design?

(3) Was the system an important one, however, representative of a major undertaking or dedicated to an important military or commercial mission?

The systems identified below yielded answers respectively of (1) No, (2) No, (3) yes.

The systems have been classified into four groups.

Base Distribution

Two major efforts in this group:

(1) The Pentagon AMPS-P, and Fort Richie AMPS-R systems.

(2) The Navy LDMX/NAVCOMPARS complex.

The AMPS system, constructed several years ago by Burroughs utilizing D-825 equipment, serves the distribution of messages to Pentagon offices. The Navy LDMX/NAVCOMPARS system will represent a significant integration step for Naval base distribution systems by combining message switching with the NAVCOMPARS programs. Both of the major functional and performance characteristics of these two systems overlap with those of the AMME system reviewed earlier.

II-29

General Store and Forward Switching Systems

Nine major efforts in this group:

(1) TADSS Tactical Automatic Digital Switching System.

(2) FAA/WMSC Federal Aviation Administration Weather Message Switching System.

(3) FAA/AFTN FAA.

(4) 497L (see below).

(5) SATIN IV Strategic Total Integrated Network.

(6) Western Union International Leased Channel.

(7) NABPARS.

(8) British Post Office.

The TADSS Tactical Automatic Digital Switching System was among the first fully automatic mobile military message switching systems. The functional and performance features of this system have since been greatly overshadowed by the more contemporary TRI-TAC engineering development design which was reviewed earlier.

497L system represented the early attempts to define a message switching system dedicated for the highly restricted Y community. It contained critical and extensive security control and validation procedures but otherwise functioned as a straightforward S/F switch. The functional requirement of this community of users was fully integrated into the AUTODIN system with little additional significant requirements on processor operations or resources other than program space.

WMSC (Federal Aviation Administration Weather Message Switching Center). This system was originally proposed for inclusion in the baseline study for two reasons. (1) It was another representative of a multi-computer complex architecture with some unusual features that were required because of the very high delivery multiplicity. (2) It is more properly a real-time information and storage retrieval system than a classical message switching system.

While the general knowledge of this system had had a tenuous, diffused, impact on the baseline, it was eliminated for detailed consideration because the methods used for message handling appear not to be reconcilable with military accountability, the traffic is repetitive and a newer report can overlay

an old report of the same type without compromise.

AFTN (Aeronautical Fixed Telecommunications Network - FAA). The FAA-AFTN switch is really a representative of many similar switching systems now in operation throughout the world. AFTN switches of almost identical requirements have been implemented on widely varying computers. As a class, they are not totally representative since most of them use only a limited number of different formats and line procedures. Beyond the initial investigation, it was felt that there was nothing unique worth expending effort on, that had not been covered in a more appropriate manner under other systems.

The SATIN IV system for the Strategic Air Command is still in the preliminary design phase. The preliminary design document illustrated applications of message switching functions to a number of small scale switches whose performance and functional requirements are overshadowed by other systems reviewed in this paper.

The NABPARS system network was never implemented. A study of available documentation revealed that the few unique properties called for such a plain address routing on format lines 6, 7, 8, 9 are covered in the base distribution system functions and, in fact, impact not significantly on the processor performance or design.

Western Union International, Leased Channel System. This is another commercial system of significant interest from only one point of view. It will, when completed probably handle more different channel coordination procedures than any system ever before attempted. Accordingly, early in the design phase the feasibility of using totally table driven channel control procedures rather than the more usual programmed approach was investigated. The designers backed off from this total table driven approach and are presently implementing a partly table driven and partly programmed approach. The reasons for not going with table driving all the way were two - the large increase in core space, and the difficulty of implementing it without various programming tools or assembler facilities of a specialized kind. The impact of both of these problems has been felt in the architecture sections and in the software tools.

Packet Switching Systems

Two major efforts in this area:

(1) C.T.N.E.

(2) British Post Office.

C.T.N.E. This is a packet switching network currently in operation in Spain. It has been implemented and gone into live operation with very little publicity and fuss. Documentation regarding this packet switching approach was reviewed. It is a clean approach and appears to have been influenced by SITA. It does not have anything unique to offer.

British Post Office Experimental Packet Switch. The British Post Office is currently implementing an experimental packet switched network. It is extremely complex and has an inordinate amount of overhead per packet. The intention of the British Post Office is to use this as a test bed for a few years, after which the superfluous functions, which have in the field been determined as being of marginal value, will be eliminated and the entire system revamped. The impact of this network was felt on the preliminary model of the packet switch approach. Other than that, it has no particular interest vis-a-vis the baseline study.

Special

ICMS - Integrated Circuit/Message Switch study culminated in the development of a feasibility model of a circuit and message switch. However, it stopped short of materializing into a message switching system with a full range of defined functional and performance characteristics, which is the objective of the baseline study. The ICMS study reports provided inputs to the architectural design model building and testing phase of the CPS study.

1.4 Major Operational Considerations

The following paragraphs describe major operational considerations which were derived from the applications studies and which apply to large classes of communications systems.

1.4.1 Message Accountability

One of the most significant determinants of system performance is the degree to which a communication system, taken as a whole, is accountable; i.e., responsible for the traffic that has been entrusted to its care. Accountability for traffic delivery is usually conceived as a total network policy. The implementation of that policy is reflected in the totality of hardware and software resources integrated at each switching center in the system, as well as the topology of low level and high level connectivity. Accountability requirements impact strongly on the transit time of traffic through the network, the holding time for messages or associated data at each node, requirements for on-line storage at each node, and requirements for various types of transmission acknowledgements. This is

directly reflected in processing delays, additional storage, and additional processing.

In general, the most significant computer resources affected are those of storage and acknowledgement processing. A rough estimate of the increase of these from minimum accountability to maximum accountability is in the order of from a 4- to 16-fold increase in computer resource requirements on a switch-by-switch basis.

A wide range of accountability intentions (doctrines, or goals) is used in message switching. These include: minimum accountability characterized by the ARPANET, and maximum accountability characterized by the procedures used in AUTODIN.

Accountability is, at best, seen as a trade-off between the user responsibility (user accountability) and system responsibility for delivery, and receipt and acknowledgement of messages. In a system with minimum accountability, a message sender (either human or SDA equipment) would expect to rely on the acknowledgement receipt of the message by a similar message generated by the destined addressee. If no such receipt acknowledgement is received at the message source, the options open to the source are few in terms of service support by the communications system. Typically, the message is retransmitted with significantly lessened confidence in successful delivery.

At the opposite end of the accountability spectrum, several of the systems (i.e., all of the military-oriented systems) have incorporated accountability procedures to extreme proportions. Basically, these extremes attempt to assure the following.

- All messages will be acknowledged for delivery and receipt.

- Complete traces of steps and processing given the message at every point of the system are maintained so that any "misadventures" of the message can be made known to the human managers of the system.

- The system itself includes procedures for the automatic retransmission or redelivery of the message, generated from its own storage media without having to resort to the message source to reinput the original message.

Accountability intentions are divided into two major areas: the system can be accountable for the traffic itself, and/or accountable for the history of the traffic. In the first case, the system maintains recordings of the message text at selected points (nodes) in the system and can thus reproduce

(retransmit) the necessary event journals to keep track of what happened to the message. While both types of accountability, traffic and historical, are usually provided in most highly accountable systems, they are in fact separable and may be treated independently at selected switches within a system.

1.4.1.1 Traffic Accountability

(1) A copy of the message may be obtained upon demand; i.e., a copy of the message awating delivery, or a copy of the message already delivered, is being retained for some predetermined period of time.

(2) The message is retained by the system (at one or more switching sites) until the system is satisfied that the message has been delivered and has accounted for all deliveries scheduled.

(3) This does not include assurance against duplicate deliveries, lost messages, lost receipts (acknowledgements), or misrouted messages.

(4) For an individual switching center, traffic accountability means that the switch will protect a copy of the message until it has received acknowledgement of all required deliveries of the message.

Historical accountability involves bookkeeping with respect to what happened to a message during its lifetime in the communications system.

1.4.1.2 Historical Accountability

(1) The switch records, on logically and electrically separate storage media, the history of the events that occurred to any message during its time in the switch. Traditionally, a dozen or more message events are recorded in the journal file.

(2) The switch operating system is able to reproduce a formatted report of these events on a message-by-message basis upon operator demand.

(3) The switch is able to display the status of the message in terms of deliverability, when, to whom, the precedence level, waiting time in queue, etc. Historical accountability thus assures against duplicate deliveries, lost deliveries, and/or lost receipts by keeping an audit trail of where every message came from and to where every message was delivered.

(4) Historical accountability is considered to place more stringent processing requirements on a switching complex. While it requires less total storage than storage of the message text itself, significantly more processing activity is required in terms of demand on I/O channels and access to on-line storage media for recording of the journal records. Historical accountability is frequently considered first in switching system design, rather than traffic accountability. The journal record files may be retained for a longer period of time than the message history files themselves.

(5) There is usually a minimum of historical accountability designed for transit centers using packet switching procedures.

(6) In entry-exit switching centers, and in base distribution systems, there is usually a high degree of historical accountability to serve the local subscribers.

1.4.1.3 Delivery Responsibility

The ability of a switching center to deliver traffic can best be seen from a design point of view; as a series of graded capabilities, each requiring more extreme processing operations and extended storage media. Delivery responsibility ranges from the following.

- Active deliver - in this state, all systems within a switch are working, and normal switch processing manages the delivery of traffic.

- Delivery from recovery in switching complexes that are duplexed or even multiplexed - delivery responsibility includes the ability of the switch to maintain delivery accountability in the event of loss of one or more duplexed components of the switching center hardware. In this case, recovery is usually "instantaneous" with little or no resultant loss in service to the customers.

- Delivery from restart - this implies the ability to maintain delivery accountability after total shutdown, but not destruction, of the active switch processors. The switching system itself is restarted, and all active tables and dynamic queues are refreshed from the journal, historical/reference, intransit, and ledger storage files maintained in logically and electrically separate media.

- The switch is released from delivery responsibility
  when it has processed the EOM signal for the message,
  and/or received one or more system level defined
  acknowledgement procedures from the downstream station.

Delivery acknowledgement procedures impact overall mes-
sage switch processing. Delivery acknowledgement ranges from the
following.

- Anti-acknowledgement - usually treated in a time thres-
  hold in which the absence of a NACK within a certain
  period of time indicates successful receipt by the
  downstream station. This is the least expensive in
  terms of overall traffic requirements, especially
  high level trunks.

- Acknowledgement to previous transit center - this is
  the most obvious and universal type of acknowledge-
  ment procedure, and is performed on either a block-
  by-block basis for high-level trunks or on a message-
  by-message basis for low-level tributaries or inter-
  switch trunks not operating on a high-level blocked
  basis.

- Receipt acknowledgement to entry center or originator
  - this type of acknowledgement procedure results in
  a significant increase in total character traffic
  over the high-level trunks of the system. It is
  usually associated with network accountability in-
  tentions and significantly raises the overall char-
  acter traffic on the high-level lines. Exit centers
  must bounce back acknowledgement receipt through each
  transit center back to either the entry center and/or
  originating terminal station itself.

1.4.1.4 Critical Processing Factors

There are several critical processing and procedural
factors which, when combined in various mixes, account for and
support the various levels of system accountability.

(1) Retention (storage) of messages throughout the
    system at one or more switching centers - retention
    of messages at given switching centers varies sig-
    nificantly according to the accountability inten-
    tion of the system. Message retention is volatile
    core storage is maintained until the time of im-
    mediate acknowledgement from the downstream center.
    At the other extreme, messages are retained in some
    medium of storage at a given switching center for
    a period of 1, 10, and perhaps as many as 30 days

before destruction. During this time, the message may traverse through a hierarchy of storage media from fast access (for purposes of retrieval) storage to magnetic tape storage or off-line disc packs.

(2) Recording in formatted information records (journal entries) of the events of processing for each message at a given switching center. The journaling or event monitoring of message processing also receives a wide range and diversity of processing treatment in message systems. At one end of the spectrum (for example, the ARPA IMP nodes, and the HH function processing of the SITA switches), message events are "journaled" in dynamic queues and tables maintained in core. The journal entries for a given message are active only during the time the message is in core; i.e., from the time of its receipt from the upstream station until the time of the receipt of acknowledgement from the downstream station. Thus these event journals are dynamic and provide little if any post-mortem analysis of the processing and/or disposition of the message after its "time" in the switch. In these particular cases, there is no recording of the journal events on a separate storage media. At the other end of the spectrum, represented by the highly accountable military systems, a carefully defined set of message processing events are generated and recorded on external media (sometimes replicated) for the purpose of post-mortem analysis and tracing of the message history. This includes copies of the message itself. Thus at the one extreme, once a message is delivered and acknowledged, it is gone from the switch forever. At the other extreme, the message and its associated audit trails "stay with the switch" for a designated period of time.

(3) A third factor of switching center accountability for message delivery is the manner and degree to which the system (an individual switching center) can recover from varying degrees of component failure, and still maintain the ability to deliver those messages as yet undelivered and/or maintain audit trail journals of processing activities that occurred up to the time of failure. Here again a wide range of functional capabilities is exhibited in message systems. At the simplest end, as the switch goes down all is lost; i.e., by the failed switch. The switching system may be brought back to life from an operational point of view, but all

traffic for which the system acknowledged a delivery responsibility (from a network point of view) is lost. There are significant variants and gradation of recovery accountability procedures between this rather austere extreme and the opposite end of the spectrum. At the opposite end, most military-oriented switching complexes are provided with several degrees of graceful degradation and recovery points. This involves the use of duplexed equipment, duplexed (redundant) storage of the several accountability-oriented files, and includes periodic ledgering of the active states of the switching processes. Full accountability and processing activity can be restored and maintained for all messages for which a particular switching center has acknowledged (assumed) responsibility for delivery; i.e., short of total destruction of the entire switch site.

(4) The final critical factor is the nature of the network accountability procedures and doctrines. This is the degree to which messages are maintained (replicated) at one or more switching centers in the system during the transit time of the message in the system, and in fact for a designated period after the message has been delivered to all intended addressees.

The nature of accountability processing installed across the several switching centers of a communication system is fundamentally dictated by the needs of accountability management for the system as a whole. Thus accountability intentions are initially defined on a total network and traffic management basis. This in turn dictates the nature of accountability processing and computer resources required at each of the switching centers comprising the system.

Network level accountability doctrines may be classified into five or six levels of accountability performance by a communications system. A brief review of these levels is essential to the ensuing analysis of the requirements for processing at a given switch.

Currently, the classes are roughly defined by various combinations of message priority, security, and other classifications in the absence of a specific accountability designator assigned by the message source. The assignment of accountability message designators is a concept that is only beginning to appear in the design specifications of systems that will probably be built in the near future.

As a network intention, user-level accountability involves no accountability whatsoever and therefore poses a very minimum requirement for computer resources and processing. An individual switch would indeed contain a message in some storage level until it is ready for transmission, and would indeed retain a copy of that message at least until receipt was acknowledged by the immediate downstream station. At a minimum, there might be some threshold set, such as the number of automatic retransmissions from one center to the next. Partial or full destruction of a switch would result in loss of undelivered traffic, and the sender would, according to his own procedures, be forced to retransmit the message.

Network accountability is frequently called entry-exit accountability. In essence, a network accepts a message at an entry center that maintains full accountability for the delivery of that message and the retransmission thereof until directly acknowledged by the switching center that is the last center of forwarding to the final destination unit. The exit center acknowledges, via a system level message directly back to the entry center, that the message has been successfully delivered. The entry center may in turn acknowledge the sender via a system level message. Intermediate high-level S/F centers maintain accountability only for active messages; the message to be released when acknowledged by the next downstream station. The upstream transit center may maintain historical audit trail records of the message for purposes of tracking lost, undelivered, or misrouted messages by tracing their events.

In a further step to assure deliverability of message traffic (particularly in military systems designed for a hostile environment), node-by-node accountability is achieved through full historical and traffic accountability procedures installed at each node of the path traversed by the message through the system. It is applied to all messages designated to receive this type of accountability management. The point is, the critical accountability processes of electrically and logically separate independent storage of both message contents and event journals are maintained fully at each node through which the message passes. The accountability for message delivery at a given node is released when that node receives acknowledgement from the downstream station. Full accountability procedures are invoked at each center for traffic passing through that center.

The link-by-link form of network accountability is a logical extension of the node-by-node accountability. A link, bracketed by the nodes (switching centers) at each end of the link, is considered as the accountable unit. Therefore, accountability is maintained at each switching center at both ends of the link comprising a fully accountable unit.

The message moves from link A to link B. These are bracketed by nodes 1 and 2 at the ends of link A, and nodes 2 and 3 at the ends of link B.

What occurs is that full accountability of a message is maintained in link A at both nodes 1 and 2, while at another time accountability is maintained at link B, at nodes 2 and 3. The net effect is to increase the overall storage and processing requirements of each node in the entire system. The effect of this, of course, is significantly increased system reliability and performance in the event of nodal or trunk failure. The destruction of any single node or any single link will not result in the loss of the message, nor of the history of its activities.

The ultimate accountability, path accountability (and most costly), is one in which each switching center traversed in the path of the message through the system maintains full and complete message and history accountability for messages designated for this accountability treatment. Release of either message or historical accountability would be performed upon complete acknowledgement of the message, and of its delivery, to all addressed receivers.

The previous paragraphs discussed accountability intentions on a network-wide basis. For the purposes of input to the HOL program, however, these must be translated to their impact on processing requirements, storage hierarchies, and configuration organization to an individual switching complex, and their impact on the language.

It should be understood that the orientation of accountability processing is toward individual categories of message accountability. This is distinct from the total accountability processing performed by a switching center. Indeed, a switch may be dedicated to handling only one class of accountable traffic, or the switch may be designed to handle several classes of accountable traffic. For practical purposes, a switch would have to be configured with resource expandability to the highest level of accountable traffic to be processed.

The characteristics and requirements for minimum accountability processing can be summarized by the following.

- Delivery responsibility for the switch that is usually restricted to active responsibility; i.e., as long as the switch is up and running it will deliver traffic for which it has acknowledged receipt and forwarding responsibility.

- Intransit storage medium for messages, in whatever envelope is being processed, is usually volatile core memory.

II-40

● Usually there is virtually no traffic or historical accounting, save for the active accountability maintained in the core tables, delivery queues, and channel queues.

● The switch assumes release for active delivery responsibility according to any one of several acknowledgement protocols that may be established on a network basis for high-level traffic. These range from positive block acknowledgement from the downstream center to negative logic NACK procedures. Positive acknowledgements are directed to the immediate upstream switch for acknowledgement of immediately received block.

● With minimum switch accountability, the following functions, are not present:

   (a) There is no capability for delivery or retransmission after switch failure and/or restart.

   (b) There is no retrieval after delivery; there is no capability for user-requested retransmission.

   (c) There is no message event journaling on separate storage media and therefore no capability for post-mortem tracing of message events to determine status, disposition, routing, etc.

   (d) There may or may not be a rudimentary form of message processing statistics; in terms of message (packet) counts, character counts, statistics on acknowledgement, negative acknowledgement, etc.

● With minimum accountability, Automatic Repeat Request (ARQ) is frequently included in the line protocols. This may be incorporated as a result of a reply to a NACK from the downstream station or may be a result of not having received a positive ACK within some system defined time threshold.

● Minimum accountability switching configurations then require:

   (a) No external (storage media) intransit file (except as an economy measure).

   (b) No external storage media for reference or journal file.

(c) No external storage media for the activity
ledger file.

An intermediate level of accountability can be conceived
as one or more steps in several directions of maintaining delivery
capability from some level of switching center failure. There
are several degrees of variability in the spectrum of accounta-
bility. These can be seen as:

- Degree of replication of processor and storage system
  components.

- Historical accountability.

- Traffic accountability.

A first step in the direction of increased accountability
is to provide a historical accountability for messages processed
by the switch. This is accomplished by the recording of predeter-
mined processing events for each message, written to an on-line
storage device. This can be applied in a reasonable manner to
the processing of high-level packet type messages in a forwarding
transit center. A trace of the arrival and transmission of mes-
sages is accounted for. This will add to the processing load for
each message-packet and will require a high-speed channel to the
storage device. Note that only message events will be recorded,
not necessarily the messages themselves, for historical accounta-
bility. In this construct, accountability is otherwise at a
minimum and the journal provides post-mortem event analysis. It
will not be used to provide delivery responsibility in the event
of recovery.

The next logical construct of increased switch account-
ability is to provide delivery responsibility in the event of
failure of one or both (if duplexed) of the active message pro-
cessing systems. Delivery responsibility is provided after such
failure by the inclusion of three processing procedures and addi-
tional on-line storage. These include the following.

- Recording of message events in the journal file.

- Recording of messages for which the switch maintains
  delivery responsibility on an Intransit Storage File
  (ISF) on some Random Access Storage (RAS) device.
  The intransit storage area is used only for messages
  with an active delivery responsibility. Space is
  released after the switch has fulfilled delivery.

- Some measure of status ledgering is required to
  maintain the storage structure of the messages
  (message blocks) maintained on ISF. These process

ledgers can be written as part of a directory on the intransit file itself, or on journal storage.

This scheme provides an off-line form of recovery, not the instantaneous recovery frequently required for redundant switching systems. All three of these processing components are required to achieve this level of accountability; the journal event, separate intransit storage, and ledgering of the intransit storage directory. There is no traffic accountability for either retrieval or retransmission after the switch has released itself of delivery responsibility for any given message.

The additional processing on a per-message basis, as compared to the minimum accountability of the previous section, includes the following.

- Writing of the message events to the journal file.

- Block gathering writing of message segments to the intransit storage file, and reading of the intransit storage file during the output processing of these messages.

- Writing of the intransit directory ledgers.

Traffic accountability can be added as an independent construct by the addition of the history/reference file. In this scheme, messages (message blocks) are recorded as received during the input processing portion of the processing cycle on a separate recording medium. If message blocks only are recorded, this file is used only for retrieval/retransmission upon request by searching (passing) the entire file to gather the separately recorded blocks of the requested message. This file by itself cannot be used for recovery in the event of failure since it will contain no journaled events of what messages were delivered, their status, etc.

This traffic accountability scheme would be added independently of the other accountability processes described above, or could be added in any desired mix. The additional processing resources required are as follows.

- Writing of message blocks as received in the host processor core.

- An appropriate I/O channel to the on-line storage device.

- In addition, retrieval and retransmission functions would have to be provided for tape searching, message reconstruction, and delivery and output processing.

II-43

Maximum accountability for a given switching center involves an extensive treatment of both historical and traffic accountability. These procedures are called for in highly strategic and tactical military environments. The environment is characterized by expected hostile activities in which there is a potentially high threat of message compromise, or destruction of one or more elements (lines and switching centers) of the communications system. Alternatively, the very procedures of traffic accountability support other functions such as message retrieval, which provide for user-oriented dissemination of message information.

Full message accountability procedures require extensive computer configuration resources (in comparison to minimum accountability processing), and add significantly to the overall processing requirement on a message-by-message basis by each switching center. Full accountability procedures add to the total storage requirement and add to the transient delay time of the message in the switching center.

Maximum accountability traffic is usually installed at entry and exit centers of long haul communications systems. In contrast to HIGH-HIGH trunk transmission, full accountability procedures are usually associated with local level transmissions: LOW-HIGH and HIGH-LOW traffic.

Typical characteristics and requirements for maximum accountability can be summarized by the following.

- Through a high degree of component redundancy in both active processors, and on-line and off-line peripheral storage equipment, the switching center is able to maintain not only an active delivery responsibility but also delivery responsibility in the event of the loss of one or several active and storage components. Delivery responsibility from varying degrees of recovery standpoints is a simple cost function. The greater the degree of component redundancy, the greater the capability for maintenance of delivery responsibility both active and in restoration.

- The storage medium of active delivery responsibility (intransit storage) is typically duplexed in on-line storage media. The redundant recording of messages intransit for delivery provides the typical two-level reliability back-up.

- In addition to the active delivery storage media (intransit storage), message segments are also written to electrically and logically independent

reference files for extended back-up, which is required under full system recovery procedures.

These reference files are typically tape files in which message segments are written serially as received during input processing, and blocked according to system design parameters. Each message or message segment received during input processing is written to the reference file. This is usually a highly complex formatted file containing segments of all messages received (acknowledged or not acknowledged, full messages, partial messages, stragglers, etc.).

- A significant part of full accountability processing involves the recording of activity; status ledgers onto on-line storage units. This involves the dynamic tables maintained in core, including the output queues, channel queues, and other processing status indicators. The activity ledgers recorded on on-line equipment are an absolute necessity for recovery from a failed duplex host computer, and are also used in total system restart for balancing the delivery responsibility of messages in the intransit and reference files.

- The switch assumes release from active delivery responsibility of a message (i.e., its release from intransit storage) when all deliveries of the message have been certified acknowledged by the appropriate network level accountability/acknowledgement procedures. After release from delivery responsibility, the message space on the intransit store is freed (through use of dynamic space availability tables) for reuse by incoming messages.

- Complete historical accountability is provided by the recording of processing events associated with each message entering and leaving the switching center. These events are time-stamped, formatted, and written to logically and electrically separate on-line storage devices. Frequently, the journal file is combined logically with the history/reference file. The journal records can be used to provide and prepare time-oriented history of any given message events upon demand.

1.4.2 Common Channel Signaling (CCS)

A recent development that has had a distinct impact on processor controlled switching systems is the advent of Common Channel Signaling (CCS). This impact is seen in both commercial and military systems.

CCS is a method for exchanging information between processor controlled switching systems over a network of signaling links. All signaling data, including the supervisory and address signals necessary to control call set up and takedown, is exchanged by this method over the signaling links.

In the commercial world, there are two CCS conventions currently being implemented, the Common Channel Inter-Office Signaling (CCIS) system of the Bell System and the CCITT #6 system used (or planned for use) by the rest of the world. Both of these systems are very similar differing only in message format variations. Both use 28-bit signaling units with 12 signaling units to a block. Both also divide the 28-bit signaling unit into 20 data bits and 8 error checking bits.

In the area of military applications, two of the systems studied incorporated CCS capabilities; the NATO IVSN system, which required the CCS scheme to be identical to or a proper subset of CCITT #6, and the TRI-TAC AN/TTC-39 system, which required a CCS scheme totally different in format, structure, error checking, block size and data rate from any existing scheme.

Regardless of any differences in the conventions or schemes used, however, all of the CCS systems are intended to perform the same set of functions.

1.4.2.1 Advantages of CCS

CCS systems offer a number of advantages over present inband signaling techniques. The major advantages are as follows.

(1) Signaling Speed

CCS signaling between switching centers is much faster than the conventional inband systems. This allows communication paths to be set up and taken down much faster. In addition, the holding time of trunks and switching equipment is reduced leading to more efficient use of these facilities.

(2) Information Capacity

CCS inherently has more information carrying capacity than is available in conventional inband signaling systems. Per-call information such as traveling classmarks or linemarks can carry routing or control information unique to each call or message. This additional information carrying capacity allows the CCS channels to be used for network control and management as well as communications signaling.

(3) Full-Duplex Signaling

CCS systems use a separate 2-way data link allowing
signals to be exchanged in both directions simul-
taneously.

(4) Separate Signaling Channel

Because CCS systems utilize separate dedicated
signaling channels, several of the problems en-
countered with inband signaling schemes are elimi-
nated. Most notable of these are the interactions
between voice and supervisory signals such as
"talk-off" (disconnect of the talking path by voice
signals) and the occurrence of mass trunk seizures
resulting from the loss of inband signals on idle
trunks due to a carrier failure.

(5) Communications Security Control

CCS provides an efficient means of monitoring and
controlling communication security and COMSEC
devices such as the TENLEY family of equipments.

(6) Reliability

CCS systems offer the potential of more reliable
transfer of signaling information than present
inband signaling techniques.

(7) Flexibility

One of the outstanding characteristics of all CCS
systems relating to network operation and user
services is that the message formats allow con-
siderable latitude and flexibility in transmitting
all type of signaling information including signals
that might be used for future user services not yet
defined. In addition, the formats provide the
flexibility to incorporate changes and additions
to the network control and management messages as
network structures and management philosophies
evolve and mature.

1.4.2.2 CCS Impact on Systems

CCS systems are essentially synchronous mini-packet
switching systems. As such, when they are applied to a communi-
cations network they superimpose a mini-packet switching function
on the network. Each of the switching systems within the network
must be equipped to handle the additional processing tasks and

load of the CCS system.

1.4.3 Network Control and Management

The advent of CCS in recent (and future) systems has provided the means for incorporating automatic (or semi-automatic) network control and network management.

In the military systems which will incorporate CCS methods, it is estimated that 5% or less of the CCS bandwidth will be required for communications signaling. This being the case, both systems, the NATO IVSN and the TRI-TAC AN/TTC-39, will utilize at least a portion of the remaining bandwidth to incorporate network control and management systems.

1.4.3.1 NATO IVSN Network Control System

At the time of this writing, there have been no documents published which define in detail the IVSN network control system. There are, however, indications that eventually the system used will be a hierarchical system. The plan appears to be the gradual incorporation of Network Control Points (NCP's) into the network and, as the network is expanded, to evolve into a stratified command structure with distinct levels of control and management.

1.4.3.2 TRI-TAC Network Control System

The TRI-TAC network control and management plans are well defined and have been published.

The management and control of tactical communications systems will use the Tactical Communications Control Facilities (TCCF) which are a family of equipments specifically designed for the purpose.

TCCF will provide automated management information and control by rapidly acquiring, processing, and displaying data to appropriate level decision makers and rapidly disseminating decisions to appropriate levels for implementation and execution. The hierarchical levels of management and control will each have identifiable functions, responsibilities, and authorities. The levels of management and control are identified as:

    a. Communications System Planning Element (CSPE)

    b. Communications System Control Element (CSCE)

    c. Communications Nodal Control Element (CNCE)

    d. Communications Equipment Support Element (CESE)

1.4.3.2.1 <u>CSPE</u>

The objective of the CSPE is to achieve optimum use of communications resources in meeting tactical requirements. It provides the means for broad systems planning, engineering, and overall systems management of the component communications systems at the appropriate levels of command.

Within any given hierarchical structure of management and control there is only one designated level which performs all CSPE functions impacting the entire structure. The CSPE estimates requirements for communications services; manages available resources to satisfy appropriate level tactical communications requirements; determines the initial system configuration; specifies locations and parameters for interfacing under the guidelines of the applicable military standards; prepares communications maps, diagrams and standard operating procedures; provides long-term management of COMSEC resources; and, ensures continuity of management and control to meet changing communications requirements.

Major CSPE functions:

a. Maintain a file of data on available resources.

b. Collect, store, update and analyze all requirements within the tactical area of operations for communications services within the tactical area and for extension of those services to and from the DCS.

c. Maintain current and planned force deployment posture so as to determine population density of requirements.

d. Plan, determine, engineer and analyze initial system configuration and interface requirements for use of, or extension through, other systems.

e. Develop implementation plans.

f. Issue taskings.

g. Monitor progress of implementation and testing.

h. Monitor overall system performance.

i. Determine mission satisfaction.

j. Establish network priorities.

k. Analyze the need for system reconfiguration and priority restructuring.

l. Monitor and update requirements.

m. Frequency management and allocation as appropriate.

n. Plan for system expansion, extension, compression, and/or reconfiguration.

1.4.3.2.2 CSCE

The CSCE provides dynamic control of large subdivisions of the communications systems. It is normally located at component levels of command immediately subordinate to the CSPE and at other locations within component forces where required for management of component and/or joint systems. The CSCE, assisted by automation, will provide real or near real-time management to maintain optimum system effectiveness. The CSCE has prime responsibility for the establishment and maintenance of a data base and provides methods for its updating and revision. These updating methods will include reports from lower and adjacent levels of control. The CSCE prepares installation orders to implement CSPE system planning and engineering, performs traffic engineering, controls COMSEC resources, makes adjustments in system configuration dictated by operational considerations, and resolves day-to-day troubles and system limitations as they arise. It is responsible for maintaining current operational status of major items of equipment and system performance. It provides immediate dynamic response to real-time needs and sends reports to, and assists the CSPE in the solution of longer term communications system needs.

Major CSCE functions:

Implementation and Installation

Prepare installation orders to subordinate levels which will implement CSPE planning/engineering actions for:

a. Installation/restoration actions and priorities.

b. General site locations.

c. Radio antenna orientation.

d. Number and distribution of essential access and trunking circuits.

e. Modification of communications links.

f. Interface facilities.

g. Switching, store and forward, and trunking equipment.

h. Other necessary implementing instructions.

Monitoring

Maintain, analyze and display system status information.

a. Traffic data.

b. System/equipment performance data.

c. Equipment restoral status.

d. Grade-of-service.

e. Outages and backlog situations.

Traffic Control Management

a. Routing control.

b. Trunk barring.

c. Line load control.

d. Trunk/link direction.

e. Control of queues.

Transmission System Routing Control

a. Reallocation of satellite and airborne radio relay channels.

b. Allocation of dedicated circuits.

c. Inter-nodal routing/connectivity.

d. Number and distribution of trunking circuits.

e. Restoration priorities.

Reporting

a. Incoming status.

(1) Installations.

(2) Equipment (on-line and spares) status.

(3) Availability of personnel.

(4) Traffic volume.

(5) Circuit/system quality.

(6) Other reports from CNCE's and other CSCE's.

b. Outgoing status.

(1) Circuit/system quality.

(2) Fault isolation and corrective action.

(3) Equipment (on-line and spares) status.

(4) Circuit and link installation/deactivation.

(5) Inter-nodal routing/rerouting/restoral.

(6) Update information for the CSPE data base.

(7) Other reports as appropriate.

Record Keeping

Management of COMSEC Resources

Directory Control

1.4.3.2.3 CNCE

The CNCE is subordinate to its designated CSCE, exercises management and technical control over its associated subordinate activities, and coordinates with other CNCE's. In the hierarchy of control levels, the CNCE functions as the nodal manager and the point of interface between the transmission subsystem, and the switching subsystem, interfaces users with the system, and has the physical, electrical, and man power capabilities to perform these functions. The CNCE is the point of electrical interface between component, joint and/or other communications systems. It is located at selected nodes (based on density, complexity and configuration requirements). It provides the CSCE with transmission system performance information and directs the appropriate subordinate activities to assure optimum system efficiency, continuity of service, and trouble correction.

The CNCE coordinates nodal activities to ensure proper balance in the application of resources, performs system/link/ channel quality control, performance monitoring, and testing, and within assigned capabilities, analyzes the data for fault isolation, determination and initiation of immediate corrective action. In addition, performance data is sent to the CSCE for further systems and trend analysis. Additional information for analysis comes from the CNCE's subordinate activities. The CNCE is also responsible for providing the CSCE with current status of on-line and spare equipment as well as the corrective action being accomplished to isolate and correct troubles. It is the primary fault prediction and isolation facility. The CNCE's process of confirming and implementing alternate routes may be automated. Semi/fully automated testing for both analog and digital systems is desirable.

Major CNCE functions:

Management and Technical Direction

Exercise management and technical direction over subordinate activities and, as appropriate, other CNCE's.

Implementation and Execution

Respond to instructions from higher levels.

Line Conditioning and Interface

a. Provide equipment necessary to line condition analog and digital circuits for optimum performance purposes.

b. Provide interface capabilities for: DC circuits, VF circuits, DC to VF conversion, analog to digital conversion, etc.

c. Circuit activation/deactivation.

    (1) Receives signal orders from the CSCE.

    (2) Directs the appropriate subordinate activities or subscribers to execute the signal orders.

    (3) Monitors the circuit activation/deactivation.

Technical Coordination

a. With the CSCE.

b. Between CNCE's.

c. With, and supervision over, subordinate activities.

Performance Assessment and Fault Isolation

a. Conduct in-service and out-of-service quality control, performance monitoring, and testing to detect indications of system degradation or equipment failure.

b. Conduct fault isolation directed toward intra-nodal, inter-nodal and extension facility (DCS, allied, etc.), transmission systems.

c. Refer fault isolation and corrective actions to its appropriate subordinate activities.

Reroute/Restore

a. Circuits.

b. Multiplex groups, super groups, PCM/TDM groups, etc.

c. Update nodal tables and directives.

Reporting

a. Incoming reports pertaining to:

(1) System and circuit activation/deactivation.

(2) Operational resources commitments.

(3) Trouble conditions.

(4) Performance data.

(5) Testing.

(6) Facility status.

b. Outgoing reports pertaining to:

(1) Trouble status of circuits, equipments and systems and restoration activities.

(2) Activation/deactivation of circuits and systems.

(3) Coordination with other control facilities.

(4) On-line and spare equipment status.

(5) Performance data.

(6) Testing.

(7) Facility/link/circuit status.

Record Keeping

Maintain appropriate records pertaining only to the current deployment.

a. Circuit record. Orders and technical directives regarding circuit connectivity, electrical and physical connections at the CNCE, nodal facilities for which it is technically responsible, and the availability of circuits and equipment at the node.

b. System and circuit status records. Maintains status of systems and circuits terminating at the node.

c. Nodal data base update (directories, tables).

1.4.3.2.4 CESE

Each CESE controls only that equipment and circuitry peculiar to itself. Its functions will be performed by equipment maintenance personnel and/or communications terminal operators using equipment organic to that facility. Insofar as it relates to the transmission subsystem, the CESE supports and responds to all management and control functions initiated by the CNCE. In addition, the CESE is capable of autonomous management and operation of all or part of its associated terminal, i.e., switch, multiplex, radio, etc. Whether operating under the technical direction of the CNCE or operating independently (only for those equipments not connected to a communications nodal point) the CESE performs equipment and channel quality control, performance monitoring and testing, equipment fault isolation subscriber equipment and channel substitution/restoral, preventive maintenance and reporting. The CESE, when unmanned, will provide equipment performance information to its designated CNCE or CSCE. This information would normally be the result of equipment self-test features and/or remote sensors.

Major CESE functions:

Technical Coordination

a. With and by direction from its associated CNCE or CSCE.

b. With other CESE's under direction of its designated CNCE or CSCE.

Performance Assessment

a. Perform in-service and out-of-service quality control monitoring and testing as required to assist the CNCE in its system performance assessments.

b. Perform subscriber loop (access line) tests to assure operation within acceptable parameters.

c. Perform other test and monitor functions necessary to facilitate optimum CESE operation.

1.4.3.3 General Comments on Network Control

The entire concept of automated network control is relatively new and, as such, is just now being introduced into the commercial networks. As more experience is gained with actual network control systems, the philosophies and technologies incorporated will more than likely change. To say at this time that the complexity and operational problems of network wide automatic control and management are not well understood is an understatement.

It should be noted that, although the tactical network control systems have been planned in detail, none of them have ever been implemented. It is anticipated that many of the features and capabilities of the various subsystems will change radically as more is learned about the real-world problems of network control. These changes will have distinct impacts on the CPS's of future deployments.

1.4.4 Dynamic Trunk Allocation Techniques

During the last two or three years several schemes for multiplexing various types of digital traffic on to wideband transmission facilities have been proposed. In each case, the purpose has been to increase the efficiency of usage of the facilities. These schemes are methods by which digitized voice, packet message traffic, narrative/record message traffic and other forms of digitized traffic can be transmitted over the same transmission facilities by dynamically assigning specific bands or slots to each class.

At the time of this writing, these systems are being studied and their impacts analyzed. In some cases, prototype hardware is being developed so that accurate simulating and tests can be performed.

One of the major study efforts being performed in this area is the Slotted Envelope Network/Digital Access Exchange (SENET/DAX) study being performed by GTE/Sylvania under the auspices of the Defense Communications Agency.

The SENET/DAX concept is based on utilization of a constant period, self-synchronizing master frame throughout multi-node networks utilizing a diversity of trunk communication rates and techniques, in order to serve a multiplicity of diverse subscribers. These subscribers in some cases (digitized voice, video, and facsimile) require essentially virtual connections between each other. In other cases these subscribers require essentially packet and/or message switched non-virtual connections.

The self-synchronizing feature of the SENET/DAX concept is based on utilization of a Start-of-Frame marker indicating the beginning of a series of constant period frames. The same constant period master frame will be utilized whatever the trunk transmission rate of the particular trunks between nodal switches may be. Following the Start-of-Frame marker, each subsequent master frame will contain slices of virtual channel connections dynamically allocated to voice/video/facsimile subscribers. This Class I region of each master frame will be allocated and maintained in accordance with link maps maintained at each end of each link. Changes in these link maps will be coordinated via CCS messages. Following the Class I region of virtual connections will be a Class I/Class II marker between those regions which will permit both a double check of the Class I allocation frame maps, and which will also facilitate efficient utilization of Class II. The Common Channel Signaling (CCS) messages required to initiate, allocate, coordinate and terminate Class I messages will themselves be handled as Class II messages. The precedence of the CCS messages will reflect in some instances that of the Class I call concerned, and in other instances will reflect the step in the connection process currently being coordinated by the nodal switches.

Class II data will be handled as packets, including: interactive, query/response, data base update, narrative/record, and non-sensor bulk data (sensor bulk data is recommended to be carried utilizing Class I virtual connections).

Idle link capacity may be synchronously filled with transmission bits, or transmission sequences, or the master frame could be treated as an essentially non-synchronous super packets,

as appropriate. However, as it is anticipated that in a military
application, significant proportions of both Class I and Class
II calls will be encrypted. It is probable that the master frame
will be synchronously filled with synchronous character sequences.

The Slotted Envelope Network (SENET) concept will permit
an effective universal communications network by dynamic alloca-
tion of variable size virtual connections for voice, low speed
video, non-sensor bulk data and forward-error-corrected facsimile
transmissions, and by utilizing the remainder of the system
capacity in each master frame for non-interfering packets of
interactive, query/response, data base update, narrative/record,
and non-sensor bulk data transfers. The Class II data packets
will be dynamically allocated as a function of the available
space in each master frame.

The implication of the SENET/DAX system is that there will
no longer be the classical distinction between circuit switching
and message switching, that the future switching systems will
handle all types of traffic. This carries with it the further
implication that each communications system has the capabilities
of circuit switching and message switching.

2.0 COMMUNICATION SYSTEMS FUNCTIONAL BASELINE

2.1 General Comments

During the course of the applications studies, a wide variance in the hardware/software allocation of functions was found. In all cases, it appeared that the primary objectives of the specific allocation of functions to either hardware or software were to provide system flexibility, the capability for rapid system reconfiguration, increased throughput capacity by off-loading routine functions from the central processors, and to achieve cost effective systems and system control.

In general, it was found that routine logic functions which were identical in every application of a particular system regardless of how the system was configured were performed by hardware subsystems or satellite processors, either Read Only Memory (ROM) driven or wired-logic driven. In each case, the function could be performed by the central processors but at the expense of processing time, throughput capacity, and additional program storage capacity. Because these functions never varied from application to application of the particular system, no restrictions on system flexibility were imposed by allocating them to fixed program implementation.

Certain system functions were found to be more efficiently performed as a combination of hardware and software operations. In these cases, the hardware functioned as an interface to the real-world environment, conditioned the information which was to be processed to be compatible with the processor logic levels, and performed a measure of pre-processing. Combining hardware and software to perform these operations provided maximum system flexibility with minimum processor loading.

It was found that those system functions which were based on parameters that varied from application to application of a system, or that had to be capable of being changed within a particular application, functions such as terminal classmarking or line marking, were implemented entirely in software. This resulted in maximum system flexibility with minimum hardware and allowed rapid changes in the system's configuration to be implemented at any site.

The system functions derived in this section, however, are described without regard to any method of implementation. They are, in fact, described in terms of the mission which the system must perform, and include all operations performed by that function whether or not they are "normally" allocated to hardware or software.

The only operations which are assumed to be performed by hardware in all cases are analog-to-digital (A to D), digital-

II-59

to-analog (D to A), and signal level conversions.

To preserve the generality necessary to compare the functions derived from the baseline studies of both circuit and message switching systems with various processor architectures, it was mandatory that the functional analysis be completely divorced from previous methods of implementation. It is for this reason that the names applied to the functions are, in most cases, new. These changes in nomenclature were deliberate and were made in an attempt to eliminate the tendency on the part of those who are intimately familiar with switching systems to equate a specific function name with a specific method of implementation.

2.2 Communications System Services

From the analysis of the systems described in the previous section, it is seen that all modern communications systems provide three distinct services. These services are:

(1) Mission Services

These are the services provided to the personnel who use the system, the subscribers. Mission services can be divided into two major categories; (1) circuit switching services, and (2) message switching services.

(2) Operations Services

These are the services provided to the personnel who are responsible for the operation, support and maintenance of the system.

(3) Network Services

These are the services provided by each switching system of a communications network to the personnel who are responsible for the control and management of the entire network.

All of the systems studied incorporated the first two services to some extent. The advent of common channel signaling techniques and network control and management philosophies has added the third service to the most recently deployed or planned systems.

In the following paragraphs of this section, each of the services is broken down into a set of distinct functions which, together, constitute the functional requirements of the service.

It should be emphasized that while it is possible to divide a communications system into an abstract set of three services

(and to divide each service into an equally abstract set of distinct functions), it is never possible to completely separate the hardware and software of an operational system into three distinct groups corresponding to the three services. Rather, it is found that for reasons of economy, efficiency, reliability and maintainability, the interrelationships of the services and their functions are extremely diffuse. Thus in a large communications system it will be found that all of the on-line processing units may be performing some of the functions involved in each of the services. On the other hand, in a small system, a single on-line processing unit may be providing all of the services.

The division of a communications system into distinct services served the very necessary purpose of reducing the complexity of a communications system to manageable and visualizable units which can be defined, examined and analyzed in detail. The further reduction of the services into distinct functions provided the degree of generality necessary to compare the efficiency of various processor system architectures to the communication tasks. The purpose being to determine which processor system architecture and which processor unit architecture best solved the overall communications processing problem.

2.2.1 Mission Services

The mission services have been divided into two major categories; circuit switching services and message switching services.

It is no longer possible to categorize circuit switches as analog transmission systems and message switches as digital transmission systems. The gradual conversion of both military and commercial communications systems to all digital systems makes these old definitions obsolete. Further, the inclusion of common channel signaling and network management and control into circuit switching networks imposes a degree of message processing capability on each circuit switch in the network. Indeed, the processing load of a 1000 line circuit switch which is also acting as the gateway node switch to a network control center will be completely dominated by the message processing functions associated with network control. The trend toward switchable lines and trunks instead of dedicated lines and trunks in message switching networks has imposed a degree of circuit switching capability on modern message switching systems.

The development of the Digital Access Switch concept in which all forms of traffic, digitized voice, narrative/record, packet, etc., are handled by the same switching system in order to allow a single set of network facilities to carry all forms of traffic tends to remove completely the designations of "a

circuit switch" and "a message switch" and replaces them with "a communications switch".

For these reasons, the real distinction between circuit switching mission and message switching mission lies in the nature of the communications path provided by each.

The circuit switching service provides real-time (virtual) communications paths between interactive end terminals such as telephones, FAX terminals, or certain classes of data terminals. As such, with circuit switched communication paths, the system appears completely transparent to the call during the traffic (or information exchange) phase of the call. In message switched communications, on the other hand, the system is not transparent to the connection but appears, rather, as a variable delay line. That is, that due to the procedures and protocols of message switching, the system is never completely transparent to the message traffic during the information exchange phase of the connection. The delays may be as short as a few hundred milliseconds (as with interactive packet traffic) or as long as several hours (as with low precedence store and forward message traffic).

2.2.1.1 Circuit Switching Services

In order for any communications system to provide the circuit switching service, i.e., to establish a real-time communications path between two terminals, it must provide the capability to recognize a request for service, to collect the requested directory number, to translate the directory number into specific routing information and to create a communications path between the calling and called terminal.

The circuit switching service can be reduced to a set of four distinct functions which satisfy these requirements.

     (1)   Supervision Signaling Function (SSF)

     (2)   Address Signaling Function (ASF)

     (3)   Path Control Function (PCF)

     (4)   Call Processing Function (CPF)

2.2.1.1.1 Supervision Signaling Function (SSF)

The SSF is responsible for the recognition and coordination of all requests for service (incoming or outgoing) on all lines and trunks. In order to perform this function, the SSF must be able to scan all lines and trunks and detect, time, decode and validate a wide variety of supervision signals (D.C.

closure, A.C. tones, E&M, SF, digital comma-free codes, common channel signaling units, etc.).

In systems where in-band trunk signaling is used, it must be able to recognize and coordinate the signaling of a wide variety of signaling conventions such as R1, R2, CCITT #4, CCITT #5, etc. In military systems numerous other conventions are used which are, for the most part, variations of the above.

In systems where common channel signaling is used, the SSF must be able to synchronize and coordinate the digital bit stream, achieving bit and frame sync and error detection. Therefore, the following subfunctions of the SSF can be derived:

(1) Terminal scanning

(2) Recognition of change of state of incoming supervision signals.

(3) Classmark analysis

(4) Multiple sampling (signal validation)

(5) Signal timing, both incoming and outgoing

(6) Operational decisions (determination of responses required, e.g., seize acknowledge, ring trip trip, etc.).

(7) Output supervision signals (analog, digital, comma-free codes).

(8) Common channel send/receive operations

(9) Performs bit synchronization on digital lines and trunks.

(10) Performs parity error (and/or code) checking on common signaling channels

(11) Generates some messages on common channels, i.e., acknowledges, negative acknowledges, etc.

(12) Performs error correction on incoming common channel messages, e.g., ARQ or error correction code processing.

There are certain C/S parameters which impact the complexity of each function. These are:

(1) Switch size, i.e., total number of terminals to be scanned.

(2) Traffic per terminal, i.e., the average busy hour per terminal call rate.

(3) Signaling conventions which must be interfaced.

(4) Subscriber features which can be signaled.

(5) Line and Trunk interface types.

The SSF requires access to classmark tables, memory for line/trunk status storage and for common channel buffering and processing, and a real-time clock.

The SSF provides the C/S system:

(1) Validated change of status information along with the line or trunk identification.

(2) Line/trunk status (idle, busy, out-of-service, etc.).

(3) Time-out information (error or malfunction).

(4) Verification of successful common channel out-going signaling.

(5) Error-free incoming common channel messages.

(6) Warning that common channel NACK threshold is exceeded.

The SSF function requires the following information from the system:

(1) Line or trunk identity for all outgoing signaling directed by the CPF.

(2) Properly coded, formatted, and validated message blocks and common channel identity for outgoing common channel messages.

(3) Classmark table updates or changes.

(4) Threshold (NACK) level changes.

2.2.1.1.2 Address Signaling Function (ASF)

The ASF is responsible for the detection, collection, validation and conversion of incoming address signaling information (directory numbers) into internally compatible data whether dialed or keyed subscriber initiated signals or address information generated from a distant switch. The ASF must also connvert

address signaling information received from the Call Processing Function (CPF) into address signaling compatible with distant switches where common channel signaling is not used.

Where AC or digital end instruments are used, the ASF supervises and controls the progression of the call phases until the traffic phase (information exchange or conversation phase) of the call is reached.

The ASF must be able to detect and send address information in any of the following forms:

    (1) DTMF

    (2) Dial Pulse

    (3) Single Frequency

    (4) Common-free Codes

    (5) DTMF Compelled

    (6) MF Compelled

The ASF performs two forms of signal validation. It first validates each individual digit by counting, multiple sampling and/or time integration and then checks the digit sequence against the numbering plan to determine that a valid number or service is being requested.

The ASF performs all operations necessary for the correct receipt and transmission of address digits or information. It includes the following subfunctions and operations.

    (1) Performs all analog-to-digital conversions for incoming AC address signaling.

    (2) Performs all digital-to-analog conversions for outgoing AC address signaling.

    (3) Performs all digital signal level conversions for incoming and outgoing address signaling.

    (4) Validates all incoming address signaling (analog or digital including comma-free codes).

    (5) Sends dial tones (or seize acknowledge tones).

    (6) Performs all digit and interdigit timing (also make/break timing).

    (7) Is capable of operating with dial pulse, multi-

frequency, multi-frequency compelled, dual tone multi-frequency, single frequency, and comma-free code address signaling conventions.

The ASF requires the following:

(1) Access to the Numbering Plan tables

(2) Access to buffer storage memory

The ASF provides the following to the system:

(1) Validated incoming address signaling information (relative to the terminal).

(2) Verification of the correct transmission of outgoing address signaling information (relative to the terminal).

(3) Notification of incorrect incoming address signaling sequences (relative to the terminal).

(4) Status of the analog-to-digital, digital-to-analog and digit level conversion hardware devices (idle, busy, out-of-service, etc.).

The ASF requires from the system the following:

(1) Directions concerning the type of address signaling convention to be used (incoming or outgoing).

(2) The address information (numbers) to be sent on outgoing signaling.

(3) Numbering Plan table update or changes.

(4) Directions concerning the type of dial tone (or seize acknowledge tone) to be used.

The complexity of the ASF is impacted by the number of signaling convention types with which it must interface. The number of subscriber terminals for which the ASF must handle the address signaling increases the processing load of the ASF and the amount of conversion hardware which must be equipped.

2.2.1.1.3 Path Control Function (PCF)

The switching matrix in any C/S provides the means of establishing real-time communication paths between terminals. To do this, the PCF must be capable of finding idle paths through

II-66

the matrix to establish new connections, be capable of tracing existing paths to release connections no longer required and to determine idle/busy status.

Since it is possible to reduce any time or frequency division matrix configuration to a space division equivalent, the remaining discussion of the PCF will be in terms of space division matrices.

Several of the systems studied, i.e., EVS, ICCS, AN/ TTC-39, required the PCF to verify path continuity either once, when each connection was established, or continuously as long as the connection existed. This function, path verification, is particularly important in Military applications where communication security is a stringent requirement.

All the military systems studied employed a precedence feature. Subscribers are assigned a maximum precedence level at which they may initiate calls. Five levels of precedence were usually employed where Routine was the lowest and Flash Override the highest. In the event a subscriber initiated a call at a level higher than routine and that call was blocked, i.e., no idle path between the calling party's terminal and the called party's terminal existed, a path would be created by preempting calls of lower precedence.

During the call processing sequence, it is necessary to send information tones to the calling, and sometimes the called, subscriber, tones such as busy, ringback, etc. In some systems, particularly TDM systems, these tones are injected into the communication path within the matrix. Other systems provided matrix terminals that were connected to tone busses which could be shared by many subscribers simultaneously. These terminals, because they could be shared, had to be treated differently from other terminals because, for them, multiple connections were normal and not an error condition. In either case, whether tones were injected within the matrix or via specially marked terminals, a unique function was being performed, i.e., tone injection.

The diagnostic problems, finding a failed crosspoint or time switch for example, in a matrix, especially a large matrix, is extremely difficult unless a set of specific diagnostic instructions have been included in the control system. Many of the systems studied required the PCF to be capable of monitoring the matrix and to isolate faults to the lowest replaceable unit, usually a Printed Wiring Board.

The diagnostic functions were usually a set of functions which were configuration and technology dependent and which allowed any matrix failure to be isolated to a single replaceable unit.

II-67

From the above discussion, it can be seen that if the PCF is capable of performing the following set of seven functions, it can control any matrix in any C/S.

     (1)   Find idle paths (Connect A to B)

     (2)   Trace existing paths (Find A)

     (3)   Breakdown existing paths (Disconnect A)

     (4)   Path verification (Verify A to B)

     (5)   Path preemption

     (6)   Tone injection (Send Tone to A)

     (7)   Diagnostic functions

In the above list, the letters "A" and "B" refer to any two matrix terminals.

In order for the PCF to perform these functions, a matrix memory map (a connect map) must exist somewhere in the system and it must be accessible to the PCF. To find idle paths through the matrix, a map of all busy paths must exist. If this map exists, then tracing an existing path and breaking down an existing path can be performed as well as finding new paths.

The path preemption function requires an additional memory requirement, i.e., the capability of storing call precedence information relative to each existing communication path. It must be possible for the PCF to trace any path through the matrix and to determine the precedence level of that path in order for it to exercise the path preemption function.

Path verification and the diagnostic function require that the PCF be able to access any point in the matrix. For path verification, it is necessary for the PCF to be able to cause a unique signal to be injected at one end of a path and to detect its presence at the other end of the path. The diagnostic function requires the PCF to be able to exercise any crosspoint (or time slot, or frequency band) and detect its proper operation.

There are several matrix parameters which impact the complexity of the PCF. These are:

     (1)   Matrix size, the number of terminals or ports which may be interconnected.

     (2)   Matrix configuration, the number of switching

stages and the manner in which they are linked together.

(3) Connectivity patterns, the connect patterns between different matrix ports to provide all services required.

(4) Traffic loading, the Busy Hour (BH) occupancy of the matrix and the Grade-of-Service (GOS) required to satisfy the switching mission.

(5) Matrix technology, the type of devices used to create the communication paths.

The size of the matrix, i.e., the total number of matrix ports, impacts the PCF in three ways, in the number of points in the matrix it must be capable of addressing, in the amount of connect map memory it must address, and in the number of matrix operations which must be performed per Busy Hour (BH).

The configuration of the switching network has a major impact on the complexity of the PCF. As an example, consider the three matrix configurations shown in Figure 2.2-1. In the figure, all three matrices are constructed with square arrays which have N inputs and N outputs where N may be any positive integer. Any one of the N inputs can be connected to any one of the N outputs in the array.
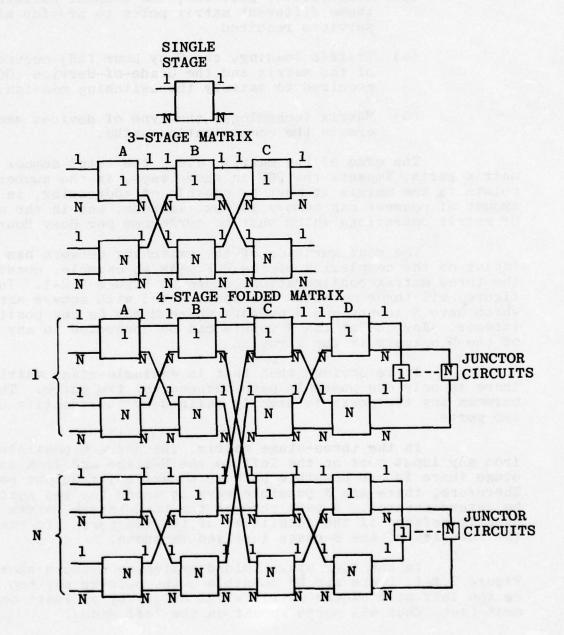
It is obvious then that in a single-stage matrix, there is only one possible path between any two ports. The path between any two ports is exactly defined by the identity of the two ports.

In the three-stage matrix, there are N possible paths from any input port on the left to the B-stage and from any B-stage there is one possible path to a particular output port. Therefore, there are N possible ways in which any two ports may be interconnected. A path through the three-stage matrix is exactly defined if the identities of the two ports are known and the identity of the B-stage involved is known.

In the four-stage folded switching network shown in Figure 2.2-1, there are N^5 possible paths between any two ports on the left side of the matrix which differ by at least one segment (note that all ports appear on the left side).

A path through this matrix is exactly defined if the identities of the ports and the B, C and D stages and junctor circuits involved are known.

From the above, it becomes obvious that as the number of matrix stages increases, the complexity of the path search

SINGLE STAGE

3-STAGE MATRIX

4-STAGE FOLDED MATRIX

JUNCTOR CIRCUITS

JUNCTOR CIRCUITS

MATRIX CONFIGURATIONS

FIGURE 2.2-1

routine of the PCF increases as well as the memory requirements of the connect map.

Complex matrices, such as the four-stage matrix shown in Figure 2.2-1, are used to reduce the number of crosspoints per port and thereby, reduce the physical size and total cost of the system.

The system trade-off in deciding the matrix configuration to be used takes into account among other things, the cost of the matrix as well as the cost and reliability of the PCF.

Connectivity Patterns

In many of the switching systems considered, access to register receiver/sender units, tone busses, and other special service units was from within the matrix. The path search routines required to connect a port to one of these special service ports was of necessity quite different from that required to interconnect subscriber ports. Wherever different types of path search routines are required because of special connectivity requirements, the complexity of the PCF is increased.

The different connectivity patterns are used because different types of connections require different Grades-of-Service (GOS's). For instance, the GOS for subscriber to subscriber connections may be 1 in 100, for subscriber to register it may be 1 in 1000, and for subscriber to trunk it may be 1 in 200. It is quite often not cost effective to design a matrix with a constant grade-of-service for all ports, therefore, systems are encountered with a variety of connection patterns.

Traffic Loading

Traffic loading impacts the complexity of the PCF in two ways, the first being the total number of crosspoints per port necessary to provide the GOS required and the second being the number of operations per second that the PCF must perform.

In general, it can be shown that the higher the per terminal traffic and the higher the GOS, the greater will be both the number of crosspoints per port for a given matrix configuration and the number of matrix operations per second during the Busy Hour (BH).

Matrix Technology

The complexity of the PCF is relatively independent of the type of matrix used. Whether it is a time, frequency or space division matrix makes little difference to the complexity, and no difference to the functions it must perform per se.

However, if devices are used which cause a consider-
able delay in the response time, devices such as electro-mechan-
ical switches which require anywhere from 1 to 100 milliseconds
to react, then the technology may add considerable complexity.
The added complexity results from a combination of the slowness
of the devices and the throughput required of large systems.
If the PCF must wait long periods of time to check that an action
has occurred then, in order to achieve the throughput, it must
be able to act independently in all parts of the matrix almost
simultaneously. This presents control problems in the area of
keeping track of all operations which are occurring and remember-
ing where to return to check on previously initiated actions.

Conclusions

The PCF is satisfied by any matrix control system
capable of performing the set of seven functions listed above
and accessing the connect memory at a rate fast enough to handle
the peak call rate.

The complexity of the PCF is directly influenced by
the size, configuration, connect patterns, and the traffic loading
of the matrix it controls. It is not impacted appreciably by
the technology used provided the timing requirements are met.

2.2.1.1.4 Call Processing Function (CPF)

The Call Processing Function (CPF) is required in a
circuit switch to coordinate the activities of all other functions
involved in completing the circuit switching mission, i.e., the
establishing and releasing of real-time communication paths.
It is necessary as part of the call processing process to analyze
and interpret the information coming from the other functions
and services so that system coordination can occur.

When the SSF reports that a particular terminal is
requesting service, it is necessary for the CPF to examine the
classmark of that terminal in order to determine what type of
address signaling will be used. A rather wide variety of ad-
dress signaling conventions are in use and are likely to remain
in use. Almost any circuit switch, military or otherwise, will
have to interface with other systems, networks, or equipment
which require various signaling conventions.

Once the CPF has determined the type of address sig-
naling convention, it must be capable of directing the ASF to
expect this type of signaling.

If subscriber access to the ASF function is through
the switching matrix, the CPF must be capable of informing the
PCF that a connection or a path is required.

It must then inform the ASF on which terminal the incoming signaling will appear. This implies that the CPF maintains the status of the ASF hardware (or terminals) and is, therefore, capable of making assignments.

In some systems, statistical records are kept of certain groups of equipment, especially pooled equipment, i.e., a relatively small group of similar equipments which are time-shared by a much larger group of equipments. The ASF hardware usually falls into this category. In these systems, the CPF must inform the operations service that an ASF terminal is in use.

The PCF is required to reply to the CPF in order for the CPF to know that the desired path was established, that no error conditions were encountered in the matrix and that no path congestion conditions were encountered. When the CPF sent the command to the PCF, it started a timer. It did this so that it would know by what maximum time it should have received a reply from the PCF. If a reply is not received within a certain maximum time, the CPF would assume that the PCF was malfunctioning and would inform the maintenance function.

When the CPF has received the reply from the PCF indicating that a path from the terminal to the ASF has been established, it starts another timer which will indicate, on time-out, that the ASF function is malfunctioning. The CPF must store information concerning the status of the call in progress because it must perform similar functions for other calls while it is waiting for the other functions to perform their operations.

When the last digit has been received from the subscriber by the ASF, it reports the digit to the CPF. The digit has been completely validated with respect to the system numbering plan by the ASF.

The CPF must then analyze the address numbers to determine if it is a local number, i.e., the number of a subscriber homed on its system, or if it is the number of a subscriber homed on another circuit switch. In the first case, the CPF must be capable of translating the digits to a matrix terminal number. In the second, it must be capable of routing the call over a trunk toward (or to) the distant circuit switch.

If it is a call to a distant switch, the CPF must examine the routing tables to determine which trunk group is involved and then examine the trunk status tables to determine if one of the trunks within the group is idle.

If all trunks in the "primary" trunk group are busy, the CPF must institute alternate routing procedures. If all trunks are not busy, it must mark that trunk for the outgoing

call.

Whether the call is local or distant, the CPF must direct the SSF to seize the line (or trunk) and commence the proper supervision signaling.

If it is a local call, the CPF will cause a path to be established and will cause the proper information tone sending sequence to be initiated (ringback to the calling party and, in some systems, ring to the called party).

If the call had been a conference call, a fact which would have been contained in the information from the ASF, the CPF would have examined the table of Conference Bridge Unit (CBU) assignments and assigned a bridge to that call. It would have determined the conferencing requirement by examining the numbering plan tables. If it had been a pre-programmed conference, a conference type in which the conference originator dials a special code which indicates that a prepared list of conferees must be called, the CPF would initiate calls to each of the subscribers on the list and cause each to be connected to the CBU and informed, via special information tones, that he was in a conference call.

If, when the CPF examined the subscriber classmark, it found that some special CRYPTO gear was required to complete the call, the CPF would initiate the commands to the operations service to coordinate synchronization and initiate commands to the PCF to connect the CRYPTO device in series with the call.

If a trunk call is to be established on a trunk in a trunk group which is supervised via common channel signaling, the CPF receives a validated, i.e., free of bit errors, common channel signaling message from the SSF. This message contains information concerning the type of call, the type of originating end instrument (in some cases), the trunk identification on which the call will arrive, and the calls destination. The CPF must be capable of examining each field within the message and interpreting its meaning.

If the destination of the call is via another circuit switch, the CPF must format a common channel message to that switch which contains the same type of information, including routing indicator information.

In order for the CPF to properly perform its functions, it must have correct information in the classmark and other tables, and a real-time clock source for timing operations.

On local-to-trunk calls, the system may be required to time the conversation phase of the call for billing or traffic metering purposes. Many of the commercial and private

II-74

systems studied incorporated this feature. The CPF, in this
case, receives notification from the SSF that "answer supervi-
sion" has been detected. It then sends a logging report to the
operation service which contains the call identification and
the start time. When release of the call is detected, the CPF
sends another logging report containing the call identification
and the time of release.

In summary, the CPF performs all operations necessary
to coordinate the activities of all other functions to success-
fully accomplish the circuit switching mission. It includes the
following subfunctions and operations.

(1) Performs all call routing and automatic alter-
nate routing.

(2) Performs (or coordinates) all security functions.

(3) Formats all common channel output messages.

(4) Formats all commands to the PCF and analyzes all
replies.

(5) Maintains all tables: classmark, directory,
call status, numbering plan, etc.

(6) Performs all digit translations and analysis.

(7) Performs, or coordinates, all feature implemen-
tation.

(8) Formats all commands to the ASF and analyzes the
reply information.

(9) Performs timing and time-out detection opera-
tions.

(10) Performs logging (time-in, time-out) operations.

(11) Controls the injection of some types of informa-
tion tones.

This function, CPF, requires access to the following:

(1) Access to all other services.

(2) Access to all other functions of the mission
service.

(3) Access to all tables.

(4) Access to data and program memory.

This function provides the following to the other functions:

(1) Overall control of the circuit switch operation.

(2) Provides statistical information to the operations service.

(3) Provides trunk assignments for outgoing calls.

(4) Provides the SSF with correctly formatted outgoing common channel messages.

(5) Provides logging information to the operation service.

(6) Provides the maintenance function with error/status reports.

(7) Provides outgoing address signals to the ASF.

This function requires the following from the other functions:

(1) Validated incoming service requests from the SSF.

(2) Validated incoming address signaling information from the ASF.

(3) Validated common channel messages from the SSF.

(4) Equipment status updates from the SSF, ASF and operations service.

(5) Real-time clock information from the operation service.

(6) Synchronization information from the operation service.

(7) Table change (or update) information from the administration function and network service.

CPF Complexity

The complexity of the call processing function is influenced by the call rate, the number of calls which must be processed per unit time, and the number of service features the

II-76

system provides. Switch size, per se, has no direct effect on the complexity of the CPF. That is, a large system which handles very low per terminal traffic and provides very few subscriber features results in a CPF which is considerably less complex than a small switch with very high per terminal traffic and a comprehensive complement of subscriber features.

Very few subscriber features require hardware equipment. A notable exception is the conferencing feature which requires specially designed hardware and is limited to the amount of hardware provided and how this hardware is organized.

Both the call rate and the subscriber features impact the speed at which the CPF must operate without incurring undo delay in servicing the calls. The call rate determines the number of call processing chores which must be handled per unit time and the subscriber features determine the complexity of these chores.

Table 2.2-1, below, is a fairly comprehensive list of the types of subscriber features which can be provided by a circuit switching system.

<u>TABLE 2.2-1</u>

(1) Station-to-Station Calling/Numbering Plan

(2) Incoming Calls

(3) Direct Inward Dialing (DID)

(4) Restricted DID Stations

(5) Outgoing Calls

(6) Station Restriction - Class-of-Service Arrangements

                                                     - Outgoing Trunk Routes

                                                     - Incoming Trunk Routes

                                                     - Internal Routes

(7) Transfer/Consultation

    - Attendant Controlled Transfer

    - Incoming Inter-Office Calls

    - Incoming & Outgoing Tie Trunk Calls

    - Outgoing Inter-Office Calls

- Station Controlled Transfer

    - Incoming Inter-Office Calls
    - Incoming & Outgoing Tie Trunk Calls
    - Outgoing Inter-Office Calls

- Station Controlled Consultation to:

    - Internal Stations
    - Inter-Office Trunks
    - Attendant
    - 3-Party Trunk Conference

(8) Universal Night Answer

(9) Predetermined Night Answer - With Transfer

(10) Power Failure Transfer

(11) Station Hunting

(12) Intercept - Vacant Level - Vacant Terminal

(13) Distinctive Ring - Inter-Office Calls

(14) Station Diversion

    - Don't Answer, Divert
    - Busy Divert

(15) Conference - Attendant and Station Controlled

    - Meet-Me Type
    - Progressive Type
    - Pre-Programmed Type
    - Broadcast Type

(16) Automatic Call Distribution

(17) Toll & Inter-Office Route Restriction

(18) Automatic Station Camp-On & Call-Back Priority

(19) Radio Paging

(20) COAM Interface

(21) Centralized Dictation Control

(22) Voice Paging

(23) Code Call

(24) No Dial Alarm

(25) Speed Calling

  - Direct Access
  - Abbreviated Dialing
  - Compressed Dialing

(26) Call Forwarding

(27) Call Pick-Up

(28) Call Waiting

(29) Special Service Features

  - Message Metering
  - Message Waiting
  - Single Digit Special Services
  - Common Battery Lines
  - Restricted Night Calling
  - Do Not Disturb
  - Wake-Up Service

(30) A.I.O.D. (Automatic Identified Outward Dialing)

(31) Wideband Switching

  - Picturephone - No PABX Features
  - Picturephone - With PABX Features
  - 2-Wire Data
  - 4-Wire Data

(32) Multi-Customer Operation

(33) Remote Console Operation

(34) Multi-Level Precedence

(35) Multi-Level Security Classifications

(36) Secure Call Mode Conversion

(37) Automatic Line Grouping

(38) Follow-Me

(39) Roving Subscriber

(40) Secret Subscriber

(41) Malicious Caller Identification

(42) "1+" Dialing

(43) "0+" Dialing

(44) Automatic Answer and Recording

(45) Calling Party Identification (Console Type Only)

As can be seen from Table 2.2-1, a rather extensive variety of services can be performed by the circuit switching system. Many of the features listed are "dial-up" types which cause the numbering plan to be more complex. Many of the features are assigned by classmark to limit the availability of the feature to selected subscribers. This in turn, increases the complexity of the classmark tables.

Experience has shown that features which were new in older systems, in general, become standard in new systems. It can be anticipated that the systems fielded in the 1980-1990 time-frame will contain new features in addition to many of those shown above.

2.2.1.2 Message Switching Services

In order for any communications system to perform the message switching mission, i.e., to accept, process, store, deliver and account for digital message traffic, it must provide the means to detect the presence of incoming messages, convert the incoming messages into a form which can be processed and stored by the system, determine the delivery destinations, route the messages to each destination, store the messages for future reference and/or delivery, and keep a variety of records to account for each message. In addition, if a variety of terminal and trunk types are terminated on the system, a means of performing speed, code, format and/or mode conversions must be provided.

The message switching service can be reduced to a set of four distinct functions:

(1) Communications Interface Function (CIF)

(2)   Header Analysis Function (HAF)

(3)   Compatibility Conversion Function (CCF)

(4)   Message Processing Function (MPF)

2.2.1.2.1 Communications Interface Function (CIF)

The CIF receives data streams from the lines and trunks and presents error-free message blocks to the system. It also accepts message blocks from the system and presents data streams to the lines and trunks. In other words, it is responsible for the successful reception, transmission and conditioning of message traffic into and out of the system regardless of the types of lines or trunks, the speeds at which they operate or what code, format or mode is used. To perform this function, the CIF must be capable of detecting incoming message activity on the lines or trunks, coordinating the transfer of data, both incoming and outgoing, and, in general, performing all line/ trunk supervision operations.

To condition the incoming information for further processing by the system, the CIF must be able to convert the serial bit stream into parallel units of the appropriate length. For outgoing data, it must be capable of reversing this process.

Since the CIF is responsible for providing the system with error-free (i.e., free of bit errors, not content errors) message blocks, it must be capable of checking parity of cyclic check code errors and requesting retransmission or performing some other form of error correction.

Therefore, the following subfunctions and operations of the CIF can be derived:

(1)   Performs all line and trunk coordination procedures for incoming and outgoing traffic.

(2)   Performs bit, character, and block synchronization on incoming traffic.

(3)   Converts an incoming, demodulated serial bit stream into characters of the appropriate length, stripping away sync pulses and other forms of envelopes.

(4)   Detects specialized characters or character sequences and decides which are to be passed on and which are to be discarded (e.g., discarding sync characters, spurious DLE characters, etc.).

(5) Performs parity checking on a character-by-character basis and/or other forms of redundancy checking (e.g., Bisync).

(6) Converts an outgoing parallel character into a serial bit stream of the proper length, adding sync pulses and other forms of specialized envelopes.

(7) Inserts special characters where required (e.g., sync characters, duplicate control characters, etc.).

(8) Performs parity generation or cyclic redundancy field generation.

(9) Performs the detection and generation of specialized non-character bit streams (i.e., for telephone signaling or dialing, for equipment monitoring, for device controls where appropriate).

(10) Performs all speed conversions.

(11) Performs all block (or character) retransmission on outgoing traffic.

(12) Performs the buffering of incoming streams into blocks for subsequent processing.

(13) Performs the detection of specialized character sequences.

(14) Performs rudimentary format checks not performed by the Message Processing function.

(15) Performs preemption of ongoing transmissions and appropriate clean-up, where required.

(16) Controls the transmission of control sequences such as ACK's, NACK's, WBT's, etc., and other forms of machineable control messages.

(17) Performs the detection and stripping out of imbedded or appended ACK or NACK or other control sequences not needed in further stages of processing.

(18) Performs output buffering.

(19) Is capable of operating with all line and trunk speeds, codes, formats and modes.

This function requires access to the following:

(1) The Line Tables, at least that portion of them that defines the speed, code, format, etc., which allows the CIF to properly interface with the lines and trunks.

(2) Memory for incoming and outgoing buffer memory operations.

This function provides the following to the system:

(1) Message blocks without parity errors along with the line or trunk identification from which the blocks were received.

(2) Errored message blocks of certain high precedence messages.

(3) Line and trunk status information (idle, busy, out-of-service, etc.).

(4) Verification of successful transmission on outgoing messages.

(5) Warns system of line/trunk error or malfunction conditions (time-outs on message ACK's, NACK threshold exceeded, etc.).

This function requires the following from the system:

(1) Properly coded, formatted, etc., message blocks for transmission.

(2) Line or trunk identity for each outgoing message.

(3) Line Table update or changes.

(4) Message detection criteria (e.g., use of tolerant detection, intolerant detection, associated rules).

2.2.1.2.2 Header Analysis Function (HAF)

In general, all of the information concerning the disposition and handling requirements of a message which is not explicit in the Linemark Table or implicit in the system's application is contained in that portion of a message referred to as the header. The validation of header information ranges from rudimentary, i.e., it is either recognizable and sensible in its entirety or the block is discarded, to very complex, i.e., each

field is validated individually for format and certain fields are cross-checked for possible conflicts between fields. In certain systems, such as the AN/TTC-39 systems, and for certain precedence classifications, headers which contain some errored fields may continue to be processed provided the non-errored fields are elements of a particular set. Whenever the header contains errors, or a particular set of errors, service messages are sent to the originating terminal or switch identifying the problem. Also, in some cases, these messages are sent to the switch supervisor for analysis and a decision as to the ultimate disposition of the message.

These operations are performed by the Header Analysis Function (HAF). It is responsible for the identification and validation of all message headers on incoming traffic and also generates validated headers for all outgoing traffic. It includes the following subfunctions and operations:

(1) Performs the analysis and validation of the following fields individually for format and collectively for possible conflicts between fields.

a. Precedence

b. Language Media Format (LMF)

c. Security

d. Content Indicator

e. Originating Station Routing Indicator

f. Originating Station Serial Number

g. Date and Time Field

h. Record Count

i. Message Serial Number - detection of sequence breaks or disorder.

j. Routing Indicators

k. Sentinels and Signals. Includes separations, hyphens, end-of-routing, etc., which must be checked for proper position and validated for correctness in accordance with the format used.

l. End of Transmission

(2) Generates and validates headers for outgoing messages containing the same types of fields as in Item (1) above.

(3) Maintains and checks message sequence number tables.

(4) Performs the service message generation required when errored headers are received.

This function requires access to the following:

(1) Line Tables

(2) Buffer storage memory

(3) Tables required for the validation procedures.

This function provides the following to the system:

(1) Completely validated message headers for further processing.

(2) Completely validated headers for outgoing messages.

(3) Notification of the receipt of high precedence traffic.

(4) Partially validated headers of high precedence messages for further processing.

(5) Notification of missing messages or message blocks.

(6) Notification of special handling requirements in the case of errored headers in high precedence traffic.

This function requires from the system:

(1) Header message blocks from the CIF.

(2) Direction and content information from the MPF to assemble output headers.

(3) Line Table updates.

2.2.1.2.3 Compatibility Conversion Function (CCF)

Most of the M/S systems studied provided message switching service to users (subscriber terminals) and trunks which employed a variety of codes, modes, and formats. In order for user terminals or trunks with non-compatible codes, formats, or modes to exchange messages, it is necessary for the system to provide conversion facilities. The Compatibility Conversion Function (CCF) performs all operations necessary to make the output messages compatible in code, mode, or format with the

receiving equipment. It also provides the code conversions neces-
sary to make the incoming message compatible with the system's
internal processing functions. It includes the following sub-
functions:

    (1) Performs one-for-one code conversions and/or
        padding up to a standard character size (e.g.,
        8-bits) on incoming messages.

    (2) Performs one-for-one code conversion and strip-
        ping of spurious bits (e.g., ASCII internal
        format padded Baudot to five level Baudot) on
        outgoing messages.

    (3) Performs operating mode conversions.

    (4) Performs alphabet code conversions.

    (5) Performs message format conversions.

    (6) Performs language media conversions.

This function requires access to the following:

    (1) Conversion Tables (or algorithms)

    (2) Buffer memory for working storage.

This function provides the following to the system:

    (1) Message blocks which are code compatible with
        the internal functions of the system.

    (2) Converted messages (or message blocks) for trans-
        mission by the system.

This function requires from the system:

    (1) Incoming message blocks which require code con-
        version.

    (2) Outgoing message blocks which require code, mode,
        or format conversion.

    (3) Directions concerning the type of conversion
        needed for each message.

    (4) Conversion Table (or algorithm) updates and
        changes.

2.2.1.2.4 Message Processing Function (MPF)

The Message Processing Function (MPF) performs both message-oriented functions and system coordination functions. All additional processing associated with a message (processing not already performed by CIF, HAF or CCF) is performed by MPF. This includes operations such as Routing Indicator (RI) detection, validation, and analysis, conversion of validated RI's to delivery instructions, etc.

In addition to these single message-oriented functions the MPF performs a task of a more strategic nature, related to the interaction of messages and the optimization of traffic. In this respect, its primary function is to schedule the utilization of available resources to optimize the system's performance.

The MPF also directs and controls all accountability related processing including keeping historical records of message, controlling intransit storage, long term message storage and journal records.

It performs all operations necessary to coordinate the activities of all other functions to successfully accomplish the message switching mission. These operations and subfunctions include the following:

(1) RI detection and validation.

(2) Conversion of validated RI's into delivery instructions.

(3) Directs the safe storage of messages on various media as appropriate to handling requirements for that message type.

(4) Directs and performs accountability related processing including the creation of historical data, converted and unconverted copies of traffic, journal information for messages.

(5) Updating of message statistics counters as required.

(6) Precedence handling.

(7) Special handling related to security and privacy classifications of the message.

(8) Preparation of message for output delivery including reformatting, editing, RI segregation.

(9) Alternate routing for outages, adaptive routing for congestion.

(10) Scheduling of processing for input lines to see to it that the SSF does not run out of incoming buffer.

(11) Scheduling of processing for output lines to see to it that a maximum output rate is maintained.

(12) Monitoring of line activity and associated message-oriented alarms as appropriate.

(13) Header parsing and identification of various control fields for: security, priority, accountability, privacy, routing, special handling.

(14) Format validation.

(15) Maintans line status tables.

(16) Performs message logging (in and out).

(17) Formats common channel messages.

(18) Performs straggler checks.

This function requires access to the following:

(1) All memory media.

(2) Line Tables.

(3) All status and logging tables.

(4) All other functions.

(5) All other services.

This function provides the following to the system:

(1) Overall control and coordination of the message switching operation.

(2) Statistical information.

(3) Message control block (character) information on outgoing traffic.

(4) Routing Indicator information on outgoing messages.

(5) Input scheduling controls.

(6) Line/trunk assignments for outgoing messages.

(7) Common channel messages.

(8) Output scheduling controls.

This function requires the following from the system:

(1) Validated message blocks.

(2) Validated header information.

(3) Input traffic demands and precedence.

(4) Validated common channel input messages.

(5) Line/trunk status information (idle, busy, out-of-service, etc.).

(6) Memory status information.

(7) Real-time clock information.

(8) Table change (or update) information.

2.2.2 Operations Services

The operations services of a communications system provide the functions necessary to monitor, control, maintain, and administer the system. These functions provide all man/machine and machine/machine interfaces, provide for system monitoring and control, provide for the maintenance and fault diagnostic procedures and provide for system start-up and system reconfiguration. In general, the operations services functions provide a very critical part of the system but are not directly involved in the primary mission of the system, i.e., the transfer of information from one point to another.

The operations services can be broken down into four distinct functions.

(1) Transmission Control Function (TCF)

(2) Maintenance Control Function (MCF)

(3) Administration Control Function (ACF)

(4) Position Management Function (PMF)

2.2.2.1 Transmission Control Function (TCF)

With the exception of commercial PBX's and PABX's, all communications systems operating in either commercial or military environments have the ability to continuously monitor and test the internal and external transmission facilities. In older systems, this function was performed with passive hardware and manually manipulated test equipment. In recent years, however, the trend has been toward automatic test and monitor capabilities with most, if not all functions, under stored program control. The developing TCCF system (refer to Section 1, paragraph 1.4.3) is an excellent example of the current plans toward fully automated technical control.

The Transmission Control Function (TCF) of a communications system encompasses all aspects of testing, monitoring, and conditioning of the transmission facilities both internal and external to the system. It is the function which is responsible for the automation of communications quality monitoring through processor control.

Within the domain of the TCF are all hardware subsystems required to interface the lines and trunks to the system, all line/trunk conditioning equipment and all transmission quality test equipment.

The hardware subsystems consists of the following:

(1) Distribution frames

(2) Patch panels

(3) EMI protection devices

(4) Automatic Emergency By-Pass switches

The line/trunk conditioning equipment consists of the following:

(1) Line termination circuits.

(2) Trunk termination circuits.

(3) Line (loop) modems

(4) Trunk modems

(5) Amplifiers

(6) Pads (resistive)

(7) Equalizers

II-90

(8) Echo Suppressors

(9) Loop-around trunks

(10) Loopback trunks

(11) Inter-matrix units

(12) 2/4 Wire analog or digital hybrids

(13) Filters

The transmission quality testing equipment consists of equipment capable of testing all or most of the following parameters:

(1) Line loss

(2) Frequency distortion

(3) Delay distortion

(4) Idle channel noise

(5) Impulse noise

(6) Path continuity

(7) Return loss

(8) Longitudinal balance

(9) Crosstalk

(10) Jitter distortion

(11) Bit frequency

(12) Frame synchronization

(13) Carrier loss

(14) Frequency response (bandwidth)

(15) Insertion loss

Also within the domain of the TCF are all "pooled" equipments not associated with the mission services functions. These equipments include:

(1) Inventory Crypto equipment

(2) Communication security equipment such as the TENLEY equipment.

(3) Special junctor circuits.

(4) Loop-around testers

(5) Automatic Routine Testers (ART's)

(6) Digital Multiplexers/Demultiplexers

The TCF not only includes the functions of the classical technical control found in older switching systems but also includes the control and monitoring of the pooled equipments and the automatic test equipment itself.

The TCF tests all lines and trunks periodically and keeps an historical record of the results. All subsequent tests are compared with previous results to determine quality trends. Thus it becomes possible for the TCF to report degradation trends and predict transmission failures.

Failure reports are generated by the TCF. The TCF also maintains channel status tables and generates channel status reports.

The TCF provides the following operations and subfunctions for the system:

(1) Provide and control line conditioning for analog and digital circuits to achieve optimum performance.

(2) Provides interface circuits, protection circuits, etc.

(3) Provides automatic or manual patch facilities and automatic readout of current patching status.

(4) Provides and controls automatic or semi-automatic test equipment for transmission quality testing.

(5) Performs transmission quality monitoring and testing.

(6) Maintains channel status table.

(7) Provides and controls pooled equipments.

(8) Performs fault isolation testing for inter-nodal and extension facilities.

The TCF provides the following information to the system:

(1) Channel status

(2) Transmission degradation and failure reports.

(3) Pooled equipment status.

(4) Current patching configuration.

The TCF receives from the system:

(1) Patching directives

(2) Fault isolation directives

(3) Monitoring and testing directives and sequences so as to avoid interruption of ongoing service.

The TCF requires access to:

(1) CPF

(2) MCF

(3) ACF

(4) All monitor control points

(5) All transmission test equipment

(6) All pooled equipment monitor and control interfaces

2.2.2.2 Maintenance Control Function (MCF)

The Maintenance Control Function (MCF) is responsible for monitoring the overall system performance and for diagnosing and isolating failed units. The major operations performed by the MCF are:

(1) System monitoring

(2) System reconfiguration

(3) System diagnostics

(4) Automatic routine testing of redundant units as well as certain on-line units.

(5) Equipment status table management.

The MCF monitors all major and minor alarm points throughout the system as well as analyzes fault or malfunction data received from other functions such as CPF, MPF, TCF, etc.

II-93

The major alarm points include:

(1) Primary power

(2) Power converters

(3) Battery back-up

(4) Environmental control units

(5) Fire detectors

(6) Cabinet overheating

(7) Main DC power busses

(8) CPU failures

Minor alarms include all other monitor points throughout the system. Typically, the distinction between major and minor failures/alarms is as follows.

A major failure is one which can result in a catastrophic failure of the system if not immediately repaired or otherwise attended to. Thus the loss of the primary power source is a major failure since it must be repaired (or replaced) within the time constraints of the battery back-up or the entire system will be out-of-service.

A minor failure is one which will effect no more than a certain percent (usually 10% or less) of the total user community. Thus the loss of a front-end processor in a message switch is a minor failure since its loss will deny service to a small set of users but will not shutdown the entire system or effect the remaining users.

In any viable operating system, all subsystems whose failure could effect more than a small set of users is replicated with an identical switchable unit. The MCF constantly tests and monitors the status of all replicated units.

The MCF maintains status tables of all subsystems as well as records of all failures.

The MCF is responsible for automatically switching over to a redundant unit whenever the on-line unit fails. It is also the responsibility of the MCF to completely isolate as far as possible the failed unit so that it cannot poison the system. The automatic reconfiguration of the system's resources to replace failed units with standby units is a major function of the MCF.

Whenever the MCF has detected a failure (or set of failures) it brings to play its diagnostic capabilities to determine what has failed. In the majority of cases the failure is obvious and quickly found. However, there are cases where the diagnostic routine must consider data from many sources in order to isolate the failure.

The MCF maintains status tables and historical records of all failures, malfunctions and alarm indications. These tables and records are continuously updated due to the automatic routine sampling and testing performed by the MCF.

2.2.2.3 Administration Control Function (ACF)

The ACF provides the means to administer and manage the system. Included in this function are the following subfunctions and operations:

(1) Data base management

(2) Program management

(3) System start/restart control

(4) Traffic metering and statistics collection

(5) Status table management

(6) Message storage control

(7) System generation aids

Data base management entails the ability to maintain, update and display the data base. The system data base consists of a massive amount of storage to define the site configuration and achieve inter-communication between services and functions within a service. The data base is subdivided into data tables which contain:

(1) Terminal and hardware physical identities

(2) Terminal classmarks or line marks

(3) Call or message precedence

(4) Directory numbers

(5) Routing Indicator tables

(6) Security levels

(7) Subscriber privileges

II-95

(8) System operating parameters such as time-out values, line/trunk load control levels, and message routing.

(9) Special features

(10) Switch location and trunk routing

(11) Numbering plan tables

Program management entails the ability to load, patch, modify, store and protect operational and support programs. In most systems, the ability to patch or modify operational programs is denied to site personnel. Program changes are achieved by loading new programs.

System start/restart entails the ability of the ACF to bring a new system on-line or to perform automatic recovery from a system outage. This subfunction includes all the clean-up operations associated with clearing connections and matrix memory maps, releasing terminals, etc., which are necessary to restart a system from an outage situation. It also includes the automatic retrieval and transmission of accountable message traffic, checking and restoring, if necessary, the data base and operational programs. The restart operation is a complex procedure which must follow a prescribed sequence if it is to be successfully completed.

Traffic metering entails the collection of a variety of statistical information concerning the number of calls or messages handled by the system, their duration, destination and routing and the usage of various equipments. Included in the traffic metering information collected are the following:

(1) Cumulative record of the total number of calls and messages generated at the switch subdivided:

- by precedence
- local calls
- interswitch calls
- holding time

(2) Cumulative record of interswitch calls and messages, subdivided by selected preference route (primary, secondary, tertiary, etc., if applicable).

(3) Cumulative record subdivided by precedence of the number of calls or messages originated by subscribers within specific line group assignments.

(4) Cumulative record of the number of preemptions on

each interswitch trunk group.

(5) Cumulative record of the number of lost calls in each precedence category.

(6) Cumulative record of the number of originated conference calls and their number of conferees.

(7) Cumulative record of the number of preemptions on local access lines.

(8) Cumulative record of operator service calls.

(9) Cumulative record of data/message calls.

(10) Cumulative record of calls routed to the operator and recorded announcements.

(11) Cumulative record of trunks on lock-out by trunk group.

(12) Common equipment and trunk group usage.

(13) Common equipment attempts by individual equipments.

(14) Call information on selected destination telephone numbers outside of the metered switch.

(15) Trunk group overflow.

(16) All trunks in a group busy - count and duration.

(17) All registers busy - count and duration.

(18) Cumulative record of Register usage - calculated average holding time.

(19) Cumulative record of intransit storage facilities used - peak storage and average storage.

(20) Cumulative record of total messages in accountability storage by precedence.

Status table management entails the recording and updating of the equipment and channel status tables from information provided by the TCF and MCF. These tables indicate for each channel or equipment whether it is idle, busy, out-of-service, on-line, off-line, standby, etc.

Storage control refers mainly to the massive storage requirements of message accountability. This subfunction directs

and controls the long term storage of accountable message traffic through a hierarchy of memory elements and provides the necessary bookkeeping and record control associated with long term message accountability. Since some systems may keep a copy of all messages for which they are accountable for as long as thirty days, various storage media are used, core for intransit storage, disc, for short term storage and finally, magnetic tape, for long term storage. Also included in this subfunction are trace operations which allow the system to find a particular message or its history.

The system generation subfunction provides aids to assist field personnel to set up the system. These are usually support programs which guide the operations personnel in the creation of data base tables, program loading and checkout, etc. Their purpose is to ensure that all information necessary to achieve system start-up is correctly formatted and inputted to the system.

2.2.2.4 Position Management Function (PMF)

The PMF provides the system with all operations services man/machine interfaces. There are four types of position consoles with which a communications system is equipped.

    (1) Switch supervisor position

    (2) Maintenance supervisor position

    (3) Technical control supervisor position

    (4) Attendant (or operator) positions.

Although in some systems one or more of the above positions may be combined into one physical console, each position will be discussed separately.

The supervisor position enables an attendant to monitor and control the operation of the system. Typically, the supervisor position involves the following:

    (1) Control of programs and data loading and system start-up procedures.

    (2) Display, print, and/or modify data base entries.

    (3) Control system operating parameters such as time-out values, line/trunk load control levels, and message routing.

    (4) Display operational status of the major subsystems.

(5) Monitor and reset the various traffic metering counters and control the limiting of lower class calls and messages.

(6) Make decisions concerning the disposition of certain messages.

In general, the PMF must provide for this position various man/machine and machine/machine interfaces and do various operations in response to inputs from the supervisor console. These functions are required to enable a supervisor to have control over the entire system operation.

The maintenance supervisor position generally requires the following:

(1) Display and printout indications of system alarms and switchover.

(2) Control of taking units on or off-line and forcing switchover to redundant units (system reconfiguration).

(3) Control the initialization, reset and start-up of units following test or repair.

(4) Initialize routine maintenance or diagnostic programs and hardware self-test modes of operation.

(5) Monitor status of the system and initiate required action in response to status changes.

As with the system supervisor position, the PMF must provide the man/machine and machine/machine interfaces and related operations for the maintenance supervisor. The maintenance position provides the system with the capability to keep the system repaired and operating, thus providing the required availability.

The technical control supervisor's position generally requires the following:

(1) Display and printout of channel status

(2) Display and printout of test equipment status

(3) Display of current patching configuration

(4) Control of test initiation

(5) Control of test equipment

(6) Display and printout of test results

The PMF provides the man/machine and machine/machine interface and related operations for the technical control supervisor's position.

The attendant or operator position provides the following services to the system:

(1) Answer any operator directed call - PMF provides a visual readout to indicate type of call and, in some cases, the identity of the originator.

(2) Extend calls to internal or external destinations.

(3) Extend calls and release operator with or without called party answer.

(4) Extend calls, release (unanswered or busy) and re-extend.

(5) Have a complete set of supervisory lamps or indicators - indicating status of the calls until answered (provided by PMF).

(6) Reenter any extended - unanswered calls

(7) Place calls on hold.

(8) Call splitting operation - talk to trunk, talk to station.

(9) Break-in on existing conversation, with break-in tone, to announce special or emergency calls.

(10) Secrecy break-in - preemption

(11) Camp-on a busy station from inter-switch trunks.

(12) Establish delayed calls - between external parties or internal and external parties.

(13) Access and control of conference equipment.

(14) Recognize busy stations visually.

(15) Busy attendant position by operating a key or automatically by removal of handset or headset.

(16) Talk inter-position (multiple attendant application)

(17) Control of message waiting service.

(18) Control of "restricted night calling" feature.

(19) Access to direct access number storage.

In order to provide the control of the attendant console, the PMF must interface with the ACF, have access to the data base and provide the logical interface (receive data from the console and provide control for light and visual displays to the console).

2.2.2.5 Operations Services Complexity

The size of a switch impacts the system monitoring by requiring larger status tables to be kept, and scanned, complicates the reconfiguration by requiring more units to be controlled, requires more statistics because of increased number of trunks and other equipment, requires management of a larger data base, requires additional interfaces to peripheral equipment and additional consoles, increases maintenance and diagnostic as new units are added to the system (with more units - more failures are going to occur), message and call queues will increase in size with the size of the switch. Control of system start-up will not be impacted significantly by size variations, and system recovery will be affected only because of the increased number of units to be controlled. The complexity of the accountability functions will not be effected directly by increase in size of a message switch. The message retrieval and tracer functions are only indirectly effected by size (complexity is increased only because of inherent increases in traffic). Typically, the number of consoles will increase with size and the maintenance and supervisor consoles will, themselves, increase in size because of the additional units involved. To summarize, the size of the switch mainly impacts the number of units to be controlled and size of the tables to be maintained and has very little impact on the complexity of the functions performed. Increased size, of course, does impact the throughput capability of the function.

The traffic load of a switch impacts the various functions as follows:

(1) Collection of on-line statistics - directly increases the number of counts, therefore, requires more messages between ACF and the other functions.

(2) Message or call queues - would increase the activity in this area - requires more messages to/from CPF or MPF, requires more control of storage of messages on the various storage devices - more storage required.

(3) Accountability functions - only affected by an increase in number of items handled (number of messages).

(4) Retrieval and tracer functions - only change is increase in number of requests received.

(5) Console control - only attendant console impacted - would require an increase in number of calls to be handled with larger queues resulting, more message to/from CPF or MPF.

2.2.2.6 Conclusion

The operations service performs all operations necessary to monitor, control and maintain the message or circuit switching system. It includes the following operations:

(1) Performs system monitoring

(2) Maintains equipment status tables and indicators

(3) Controls all automatic system reconfigurations

(4) Performs the collection of on-line statistics - traffic metering

(5) Controls switch supervisor, maintenance supervisor, technical control supervisor, and attendant consoles.

(6) Controls all man/machine and machine/machine interfaces including all code, speed or format conversions required for communications compatibility with the peripheral devices.

(7) Controls and performs all maintenance and fault diagnostic functions and procedures.

(8) Performs all accountability related storage and queueing functions.

(9) Performs traffic (and system) recovery functions and controls recovery procedures.

(10) Performs message retrieval operations

(11) Performs all tracer actions

(12) Controls system start-up procedures, including program loading, data base loading, etc.

(13) Performs all message or call queueing operations

(14) Performs data base management

(15) Performs all transmission quality monitoring and testing.

(16) Maintains channel status tables.

This service requires the following:

(1) Access to all peripheral interfaces

(2) Access to all transmission media

(3) Access to all memory devices

(4) Access to all position equipment

(5) Access to all tables

(6) Access to all error/status monitoring points throughout entire system.

This function provides the following to the system:

(1) Messages (or message blocks) from various memory devices

(2) Table update or change information

(3) Procedural instructions

(4) Maintenance/diagnostic instructions

(5) Queueing information

(6) Programs and data base information

(7) Channel status information

2.2.3 Network Services

The network services are a fairly recent addition to communications systems and thus far exist only in commercial networks. At the present time, there are no military networks which incorporate the network services. Both the TRI-TAC AN/TTC-39 tactical networks and the NATO IVSN strategic network are being designed to incorporate the network services to some extent.

The network services consist of three functions:

(1) Network Control Function (NCF)

(2) Network Management Function (NMF)

(3) Remote Maintenance Function (RMF)

The network services are services provided to the personnel who are responsible for the overall management and control of the entire network. These services consist of supplying the information necessary for optimizing the use of the network facilities and responding automatically (or semi-automatically) to their commands.

Network Control is defined as those operations which are necessary to dynamically control the flow of traffic through the network. As such, it is responsible for guiding traffic around congested or damaged areas by forcing the traffic to be routed through less congested trunk groups and switching centers.

Network Management is defined as those operations which are necessary to provide for the orderly expansion of the network or changes to the network. In other words, network management is responsible for the long term planning of the allocation and control of the network's resources. It also includes the development of the procedures and protocols to be used to optimize network usage and network changes.

Remote Maintenance is defined as the operation necessary to assist relatively inexperienced site personnel to maintain and repair the sophisticated switching equipment through interactive data exchange and remote diagnostic capabilities.

2.2.3.1 Network Control Function (NCF)

The network control function is divided into two main operations; report generation and command execution.

Report Generation

In order for the network controllers to perform their mission, it is imperative that they have at their disposal timely information concerning the traffic load and the current status of the channels and equipment in each switch. Each switch is capable through the TCF, MCF and ACF functions of the operations services of collecting the necessary traffic metering information (from the ACF), equipment status information (from the MCF), and channel status information (from the TCF). This information is forwarded to the Network Control Center (NCC) in the form of reports consisting of three types:

II-104

(1) Traffic metering reports

(2) Equipment status reports

(3) Channel status reports

Traffic metering reports are sent whenever preset traffic thresholds are exceeded or when requested by the NCC.

Equipment status reports are sent whenever a change in the status of an on-line unit takes place or whenever a failed unit has been repaired and placed on standby or active status.

Channel status reports are sent whenever a change in the status of any channel is detected or when channels which were out-of-service are restored to service.

At any time the NCC can request the current status of the equipment or channels or request the current traffic report.

Command Execution

The NCC, in response to the information contained in the reports from all switches in the network, may issue commands to control the traffic flow in a particular switch. These commands generally fall into one of four categories which are:

(1) Time-out Controls - These are readily alterable software time-out functions used in the call or message processing routines. They include common control, user and trunk circuit time-out thresholds.

(2) Traffic Load Controls - These include readily alterable software parameters by which the traffic offered to trunks by users can be controlled during peak busy loads. This is accomplished in two ways. First, by restricting less essential users from initiating traffic while handling existing traffic in the normal manner, and second, by instituting inter-switch trunk restrictions, i.e., limiting trunk usage to essential (precedence) traffic only.

(3) Trunk Directionalization - This includes readily alterable software parameters by which inter-switch trunk groups can be individually directionalized to control network traffic flow. Specifically, each inter-switch trunk groups provides two modes of operation; normal two-way traffic and directionalized traffic where outgoing calls are not allowed.

(4) <u>Routing Controls</u> - These include readily alterable software parameters for placing restraints on the routing of calls to control network traffic flow. The following parameters are included:

    a. Cancel specific alternate routes (where deterministic routing is used).

    b. Cancel all alternate routes (if applicable).

    c. Cancel specific switch codes.

    d. Switch to a back-up alternate routing table (if provided).

    e. Specification of path length restrictions (maximum number of links in tandem).

    f. Cancel all calls or messages to switches designated as out-of-service.

2.2.3.2 Network Management Function (NMF)

The information from the switches required for the network management function is supplied by the same set of reports used for network control. This information is used by the management personnel to generate long term network traffic congestion patterns which in turn are used to reallocate the network resources to provide optimum communications.

The information is also used to generate equipment and channel failure rates and trends so that channel allocations, equipment spares estimates and alternate routing plans can be made to achieve the least disruption of communications throughout the network in spite of failures.

Another way in which the report information is used by the network management personnel is in the reallocation of resources within the reporting switch itself. By examining the traffic metering information concerning the usage of common and pooled equipments, the network managers can determine when the resources of a particular switch must be increased or re-engineered (in the sense of office grading).

The network managers provide information to the switches which is of strategic value through the NMF. Included are:

(1) Classmark table update - especially concerning trunks and trunk groups.

(2) Linemark table update

(3) Routing table updates

(4) Alternate routing updates

(5) Threshold levels - traffic and transmission quality

(6) Reporting periods - periodicity of reports in lieu of specific requests.

(7) Directory number changes or additions

(8) R.I. changes or additions

(9) Drop and insert assignments

(10) Trunk assignments

(11) User privileges assignments or changes

(12) Message accountability protocols

(13) Equipment changes - additions to or deletions from equipment pools

(14) Network services routing assignments and alternate routing assignments.

2.2.3.3 Remote Maintenance Function (RMF)

The remote maintenance function is finding wide acceptance with commercial communications operating organizations. The Bell System and the German Bundespost are just two examples of communications organizations which have incorporated remote maintenance functions into their new electronic stored program controlled switching systems. The impetus for doing this is due to the unusually high reliability of these systems. The failure rates are so low that it has literally become almost impossible to keep individual on-site maintenance crews technically competent to repair the system when a major fault does occur. Their experience has been such that without expert outside assistance the on-site maintenance personnel have exacerbated the problem resulting in extended outages.

This situation has led to the semi-facetious "one man and a dog" maintenance philosophy. A maintenance man is assigned because system managers simply are not happy with unattended switches. The dog is assigned to see to it that the man does not touch the equipment.

The actual maintenance philosophy used is to staff the

maintenance crews with personnel capable of correcting minor failures and clearing minor alarms but to provide a means for an expert crew situated at a remote Maintenance Control Center (MCC) to diagnose, supervise and control the repair of a major malfunction via a maintenance order wire system and an inter-active data terminal.

This concept has several distinct advantages:

(1) The MCC personnel are exposed to enough major failures to remain technically competent.

(2) Lower grade (in the sense of technical ability) personnel can handle the routine maintenance tasks at the site.

(3) The large and extremely complex diagnostic pro-grams need be stored only at the MCC resulting in considerably less program storage resources at each site.

The main disadvantage is that the RMF must be incorpo-rated into a system during the design phase if it is to be en-tirely effective. This is due to the fact that it must have diagnostic controls and sensors built into strategic points of the system so that it can perform its function effectively. Post design ad hoc measures, while they can be made effective, are invariably more expensive and less efficient.

A properly designed MCF of the operations services pro-vides the necessary interface for the RMF.

Although none of the published plans have indicated the inclusion of the RMF into the new military systems, it is antici-pated that as the advantages are realized, it will be gradually implemented.

2.2.4 Summary of Services and Functions

The Services and Functions of C/S and M/S communications systems are summarized as follows:

I. MISSION SERVICE

(Circuit Switch Functions)

1. Supervisor Signaling Function (SSF)

2. Address Signaling Function (ASF)

3. Path Control Function (PCF)

4. Call Analysis Function (CAF)

(Message Switch Functions)

5. Communications Interface Function (CIF)

6. Header Analysis Function (HAF)

7. Compatibility Conversion Function (CCF)

8. Message Processing Function (MPF)

II. OPERATIONAL SERVICES (Common to both M/S and C/S)

9. Transmission Control Function (TCF)

10. Maintenance Control Function (MCF)

11. Administration Control Function (ACF)

12. Position Management Function (PMF)

III. NETWORK SERVICES (Common to both M/S and C/S)

13. Network Control Function (NCF)

14. Network Management Function (NMF)

15. Remote Maintenance Function (RMF)

3.0 THE ARCHITECTURE OF THE CENTRAL COMPUTING COMPLEX

3.1 General

3.1.1 Overview

This section is an exposition of the Central Computing Complex (CCC) architecture and the steps that led to the present form thereof. Section 3.2 discusses the evolution. The entire architecture and the detailed considerations associated with each feature and instruction is to be found in Volume IV of this report.

3.1.2 Goals - The Application Domain of the Architecture

Four sets of objectives led to the present form of the architecture:

    (1) Functional requirements

    (2) Range of resource requirements

    (3) Operational survival-viability

    (4) Architectural survival

Each of these aspects will be discussed in further detail below. The goal of the project was to produce a specification for an architecture that would provide cost effective processing in response to the above subsidiary objectives. Of these, only the first three were explicitly addressed in the Statement of Work. The fourth, architectural survival, and for that matter, the third, operational viability, are usually taken for granted. Their impact, however, is at least as important as the first two, and therefore will be explicitly discussed.

3.1.2.1 Functional Requirements

The detailed functional requirements are discussed in Section 2 and will not be repeated here. What is important from an architectural point of view, is not the details, but the range of these requirements, since most of the details are a matter of software design which has at best a diffuse impact on the architecture. The functional range is very broad - encompassing such diverse requirements as message switching, circuit switching, packet switching, and many aspects of tech control functions, if not tech control in toto.

The broad functional range is in response to a recognition that the boundaries between these seemingly diverse applications has gradually become fuzzy, and if trends are to be believed, will become fuzzier still in the future. Thus it is no longer a mere matter of providing a computer or computer complex that will separately serve each of the potential application areas, but one that will effectively serve a range of applications in which the characterization as, say, a circuit switch or message switch is at best an indication of dominance and polarity rather than an absolute.

Had the applications been totally separable, the task would have been easier. Each application would have demanded special instructions, idealized to that application only. The individual computers would have been simpler, and probably more cost effective than the proposed architecture. But the functional aspects of the applications were not separable, and consequently any special feature, ideal for, say, packet switching, had to be effective and useful in message switching and circuit switching as well. Because of the relatively large size of message switches and base distribution switches as compared to circuit switches, message switching, while numerically not accounting for the bulk of the installations, exerted an influence which tended to bring the architecture closer to that which is needed in a non-communications, data processing environment than would have otherwise been expected. For example: large files are the rule in message switching, the exception in circuit switching. The main memory of message switches is typically larger than that of circuit switching. Other examples are discussed in the various baseline studies.

Had circuit switching alone dominated the applications, it is likely that the architecture would have been based on a monolithic computer rather than a multi-computer complex, and the CPU would have been a 16-bit minicomputer, rather than a 32-bit midi. While such an architecture would not have been cost effective for all circuit switches, it would have sufficed for the 600 line circuit switches of the past. That it would have sufficed for future circuit switches, particularly in the all digital world, is not likely.

3.1.2.2 Range of Resource Requirements

By "range of resource requirements" is meant some measure (albeit imprecise) of the range of sizes of the expected installations. "Size" is also an imprecise characterization of a system. It involves total working memory, mass memory, number of instructions executable per second by the complex, number of channels, the power of the repertoire, the word length, etc. What is important here is not the absolute size, but the range; and more explicitly, the distribution of size. Without agonizing

over how to measure the capacity of a computer complex, it is possible to arbitrarily establish some measure of size, ranging over possible applications on a scale of 1 to 10. Conceptually, a probability distribution of the possible sizes for various installations can be drawn; with very few of size 1 and very few of size 10. The shape of this conceptual size distribution clearly affects the architecture which is designed to meet it. If the mean and mode had been crowded at the lower end, inefficiencies for the relatively few very large installations would have been accepted and a mini-computer would have been specified. If, conversely, the applications had centered at the upper end, inefficiencies at the lower end would have been acceptable. Optimizing for the average size is a poor strategy since it makes all applications inefficient. For example, the distribution could have had a peak at the upper and lower end, with relatively few installations at the middle. Ideally, a close fit throughout the entire distribution needs to be obtained. The architecture is therefore clearly influenced by the shape of this hypothetical installation distribution.

In the case of the CPS, the distribution was flat and broad. While very small circuit switches did not require much in the way of equipment, there were to be many of them. While very large base distribution switches are few, they are individually large. These two characteristics tend to flatten the end points. In fact, for all application areas, be it circuit switching, packet switching, message switching, base distribution message switching, tech control, or digital concentration, the number of installations tends to be inversely proportional to the size. This creates a tendency to flatten the distribution. The architecture, therefore, must be cost effective over a very broad range of resource requirements - comparable to the application range of commercial data processing equipment.

3.1.2.3 Operational Survival - Viability

Unlike commercial applications in which availabilities of .998 are acceptable, military communications strives for 100% availability. While this is theoretically impossible to achieve, military viability objectives increases from .998 to .9995, to .99995 have been seen. Automatic recovery and graceful degradation is the rule, not the exception. While availabilities of .99995 may seem adequate when taken on a single switch basis, when considered in the context of an entire network, which may have many switches serially interposed between users, such availabilities are marginal. While much can be and is done at the network level to improve user-to-user viability (e.g., through alternate routing, network control, etc.), ultimately, it is the serial chain from user-to-user that dictates effectiveness, and there seems to be no let up in the desire to push individual switch viabilities further yet. If, on top of failures and

malfunctions due to "natural" causes (such as hardware and software malfunctions) scenarios of national emergencies and catastrophes, a hostile environment and realistic assessments of maintenance skills are superimposed, it is found that the military has in fact not been overstating their viability requirements, but have resigned themselves to accept that which the state-of-the-art can economically provide.

Do what you will with quality control, elegant circuit design, semiconductor physics, production testing, exotic materials, environmental control, and power distribution, viability today is achieved by redundancy - by the replication of critical units. Furthermore, since viability requirements have tended to escalate with improved failure rates, no future point at which some form of redundancy will not be the major means by which viability objectives are achieved can be foreseen. Thus, if one unit of a particular resource is needed, it will be satisfied by 2 units - the primary and the standby. If N units are needed, at least N+1, possibly 2N shall be provided. While the commercial computer is rarely replicated, the military computer is (almost) always backed up. If the same availability can be achieved by N+1 replication, rather than by total redundancy (e.g., duplication, or 2N replication) it is clearly a means by which cost can be reduced, and therefore cost performance improved. The means by which adequate redundancy is achieved, and the cost thereof, is therefore a central issue in the design of the architecture.

While the replication issue is the most important, there are numerous other issues, corollary to it, that have a diffuse but nevertheless significant impact. The ability to provide on-line testing and diagnosis, the isolatability of failed units, prevention of technician induced failures, power distribution, environmental control, details of circuit and interface designs, etc., all create pressures that influence the architecture. In the final analysis, these factors pervade the entire architecture and lend to it a distinct flavor found only in military gear and communications gear as used by common carriers - it is a flavor distinctly lacking in the overwhelming majority of commercial computers.

3.1.2.4 Architectural Survivability

This is perhaps the subtlest requirement - so much so, that it is taken for granted and hardly ever explicitly stated. Simply stated it is that the architecture should survive - survive despite improvements in component technology and costs, survive despite gradual shifts in the qualitative and quantitative aspects of the applications, survive despite changes in maintenance personnel qualifications, survive despite modifications to logistics policies, survive despite the source language used for the software, survive despite the future shocks that all

technology is heir to. An architecture must be designed so that it shall not be obviated - to do less is irresponsible - to ask for less is to ask for temporary solutions and ad-hoc approaches, not for a new architecture.

What does survival mean? Merely able to work for the next twenty-five years? Clearly nothing so restrictive. It means competiveness or cost/effectiveness, based not on just purchase costs, but total life cycle costs. A vacuum tube computer is not effective today since power and maintenance costs exceeds the cost of replacement by a far more powerful modern computer. A mini-computer architecture with say an upper limit of four channels, cannot survive because it cannot grow to encompass expansion in availability, or in core or in functions that forces the interconnection of ten computers. Ultimately, and realistically, every architecture will be obviated - be it of ships, buildings, aircraft, bridges, or computers. But likewise, it is incumbent upon the architect to forestall the inevitable by building in such features as will allow the architecture to provide long term survivability.

In the case of computers, and especially those that are used in communications, be it military or common carrier, architectural survival is a more crucial and important issue. Life cycle plans of 10 to 20 years are not unusual. However, most of this gear tends to have much longer life cycles. There is a capital investment in logistics, training, and most important, software that cannot readily be discarded. The latter, particularly, is coming to dominate life-cycle costs. Software too, eventually becomes top-heavy with modifications and must be redesigned. However, the evolution of new functional requirements that demand software changes is generally slower than the evolution of technology and hardware. In most cases where software has been obviated it has been coupled to a replacement of hardware. If the hardware architecture does not survive, it cannot be expected that the software will survive.

The design of an architecture for survival in this sense is accepted, though rarely stated, design practice among all viable commercial computer manufacturers. As archtypes, IBM and DEC can be taken as representative of the most successful computer companies. IBM's introduction of the 360 series represented a dramatic (and for the users, traumatic) departure from the previous product line. The 7000 series had been a mere "transistorization" of the 700 series vacuum tube line. In addition to these two main lines, there were the small scientific 1600 series and the small commercial 1400 series. There was no commonality between these and little commonality across various elements of the 7000 series. IBM had a diverse, incoherent, limited product line; totally incapable of growth or survival. At one stroke, IBM obviated its entire product, antagonized its users by the introduction of a new architecture, a new language,

new operating system, job control language, control console - in fact new everything, almost completely unconstrained by the previous product line.

The 360 series was originally constructed around what IBM called "solid logic technology" which was a discrete/integrated circuit hybrid. Passive components such as resistors and capacitors were deposited, as were insulation and interconnections, on a ceramic substrate in the fashion of present day integrated circuits. Resistors and diodes were discrete and welded in place. That circuit technology did not survive for long, but the users were oblivious to the change, which occurred several years later. There has been a gradual shift in the product line towards LSI memories instead of core. Top of the line features have migrated downwards (compare for example the 360/50 to the 370/145). The upper end of the line has been continually expanded. All of this has been accompanied by gradual shifts in details, and minimal impact on pre-existing application software. The introduction of the 370 series by virtue of the fact that it was almost indistinguishable from the 360 series, underscored IBM's intent that this series shall survive. It might be renamed the 380 series in 1980, the 390 in 1990, and possibly a new series might emerge by the turn of the century, but it is doubtful.

DEC in a similar way, by introduction of the PDP-11, declared its new "forever architecture". Other manufacturers with varying degrees of success, have coordinated and regularized their product lines and are appearing to be moving in the direction of a "forever architecture".

Architectural survivability is not a haphazard thing. It does not just happen. It, like viability, must be designed into the system. The following characteristics, while not guaranteeing architectural survival, do mitigate towards it:

    (1)   Generous Logical Limits on Terminations. This means that the number of bits allowed for addressing units and memory are made significantly larger than would be needed by predicted applications.

    (2)   Minimization of Timing Dependencies. It should be possible to replace one unit type at a time, as economics dictate, generally to a higher speed, lower cost replacement. A totally asynchronous design is preferable. If this is not feasible, timing dependencies should be minimized.

(3) <u>Technology Independence</u>. It should be possible to replace the technology with no more than the addition of (possibly) special electrical interfaces. Logic and protocols should not be affected by component technology.

(4) <u>Escape Hatches</u>. Provisions for expansion of the repertoire, changes in word length, increase in the number of registers, and increase in the number of units should not be ruled out by inherent structures in the architecture. It should be possible to create an upwards compatible version of the same architecture which can run the programs of the old in a non-emulator mode. The idea behind all escape hatches is that they should cost nothing if not implemented, and there should be nothing fundamental in the architecture that prevents such expansion.

3.2 The Evolution of the CCC Architecture

3.2.1 The Architectural Process

3.2.1.1 General

It would be intellectually satisfying to be able to describe the process by which the architecture of a computer complex evolves as an analytical and deductive process. However, that is not the nature of the process. While deduction and quantitative analysis do play an important role, they account for, at most, half of the effort and for none of the creativity. The cornerstones of architecture are intuition and serendipity. Analysis is used to fill in the details and to provide a post-hoc rationalization of that which was determined intuitively. This is not intended to denigrate the role of analytical and quantitative processes in the evolution of an architecture, but merely to place such processes in their proper perspective.

Analysis plays the role of feedback. Intuition and serendipity leads to an idea - analysis is used to test the validity of the idea and if it should survive that test, to provide new inputs for the creative, intuitive processes to work on, leading to yet further ideas. The actual process then, is one of continual interplay, with all things going on more or less simultaneously. A discussion of that process is at best a charicature of the real thing since intuition and serendipity cannot be described, only deduction and analysis can. The discussion, tnen, of the evolution of the architecture is perforce representative, retrospective, and inadequate. Only a minority of the activity can be reduced to documentation. The "steps" discussed below are approximate at best; and the reader will
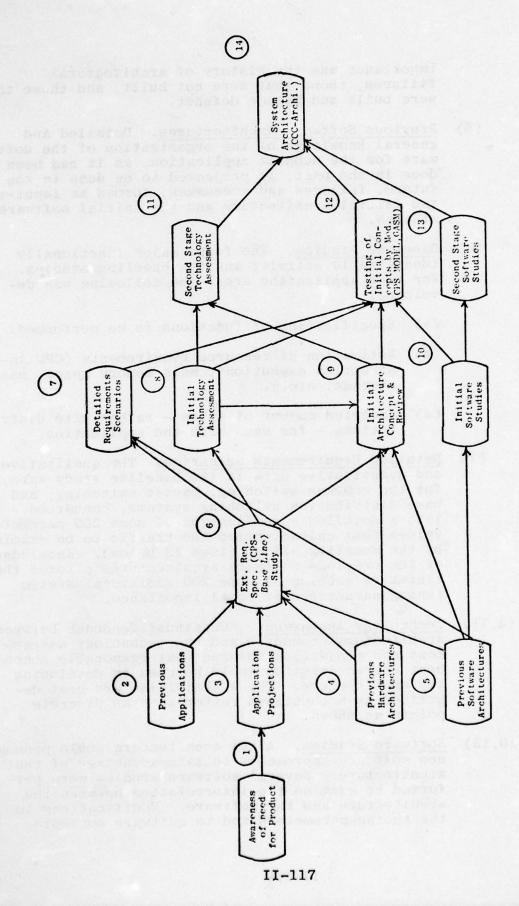
still find himself asking questions of the form "how did they get from here to there?" To answer all such questions would be to needlessly wander into an essay on the psychology of invention, creativity, and similar subjects. However, the following can be done:

(1) Show the major steps that occurred; at least those that can be identified as discrete activities.

(2) Show the rationale behind the major architectural decisions - which is to say the tests that those decisions were subjected to.

(3) Show the major architectural alternatives and the reasons why they were selected or rejected.

3.2.1.2 Steps In the Process

Figure 3.2-1 shows the identifiable steps in the process by which the architecture was devised. In reality, almost all items identified overlapped with their predecessors and successors, and almost every box had feedback to most other boxes. The process shall be discussed as if it had gone as shown in the figure.

(1) Awareness of a Need for the Product. This activity was carried out by the sponsoring agency. It is a starting point for the process.

(2) Application Projections. This too was a sponsor activity. In reality, it consisted of collating, organizing, and regularizing the numerous planning documents provided by the various potential users of the system. This resulted in specific items (reflected in a statement of work for this project) and more important, in a collection of background documents and attitudes, which collectively defined the application.

(3) Previous Applications. The requirements of known previous applications in the form of actual and projected switching systems were important inputs to the baseline study.

(4) Previous Architectures. General and detailed knowledge of previous architectures, both used in switching applications, commercial applications, and other military applications, as well as projected architectures not yet considered formed a major input to the tentative architecture, and a minor input to the baseline studies. Of equal

Awareness
of need
for Product

Previous
Applications

Application
Projections

Previous
Hardware
Architectures

Previous
Software
Architectures

Ext. Req.
Spec. (CPS-
Base Line)
Study

Detailed
Requirements
Scenarios

Initial
Technology
Assessment

Initial
Architecture
Concept &
Review

Initial
Software
Studies

Second Stage
Technology
Assessment

Testing of
Initial Con-
cepts by Mod.
CPS MODEL, GASM)

Second Stage
Software
Studies

System
Architecture
(CCC-Archi.)

Fig. 3.2-1 ARCHITECTURAL PROCESS

importance was the history of architectural
failures, those that were not built, and those that
were built and became defunct.

(5) Previous Software Architectures. Detailed and
general knowledge of the organization of the soft-
ware for the subject application, as it had been
done in the past, is projected to be done in the
future, failures and successes, formed an input to
the initial architecture and the initial software
studies.

(6) Baseline Studies. The first major functionally
identifiable activity was the baseline studies.
For each application area, the following was de-
veloped:

(a) Specification of functions to be performed.

(b) Estimation of resource requirements (CPU in-
struction executions, main memory space, mass
storage, etc.).

(c) Expected number of sites - rather site distri-
butions - for each size and application.

(7) Detailed Requirements Scenarios. The qualitative
and quantitative data in the baseline study were,
for the message switching, packet switching, and
base distribution switching systems, converted
into a detailed specification of some 200 parameter
values that characterized the traffic to be examined
by the modeling effort (item 13 below). Knowledge
of the previous software architectures allowed the
tentative setting of some 200 additional system
tuning parameters of equal importance.

(8,11) Technology Assessment. Continual feedback between
architectural concepts and the technology assess-
ment and projection assured that reasonable compo-
nent characteristics would be used in developing
the architecture. These activities are best de-
scribed as a continuum rather than as discrete
points as shown.

(10,13) Software Studies. A new architecture would presume
new software approaches to take advantage of that
architecture. Several software studies were per-
formed to examine the interrelation between the
architecture and the software. Modifications to
the tuning parameters and to software concepts

resulted from this, which modifications were incorporated in further stages of the evolution of the architecture. A split in software occurred, identifying the need for separate consideration of application software and software development tools to be associated with the architecture. As with the technology assessment, this was an ongoing activity more typified by a continuum than by discrete points.

(9)   <u>Initial Architecture</u>. The combination of the above activities led to the development of an initial architecture. This was produced as a document and distributed to project personnel and experts in the application areas. That architecture was intended to be maximal in the sense that it contained anything that had a shred of usefulness. It would of course, not have been cost effective, even if it had been feasible. The idea was to subject it to expert critique. In all, some 30 experienced system designers, programmers, and architects reviewed this document. Copious suggestions and criticisms were provided. The feedback provided by this was an important input to subsequent stages in the evolution and a highlight of the project.

(12)   <u>Testing of Concepts by Model</u>. Many concepts emerged in the tentative architecture document. The criticism served to expand some of these, to delete many, and most important, to identify a relatively small number which required quantitative evaluation in order to make a rational decision. Concurrent with the entire process (but not shown explicitly) was the development of a number of models which were used to test possible features as to their qualitative desirability and as to their quantitative impact. Wherever possible in the development of the models and the associated modeling tools, provisions were made to allow the testing of questionable features. Not all features could be tested explicitly. Some had to be approximated, and some could not be tested at all. Some will still require verification and testing in future phases of the project (as detailed in Volumes III and IV).

Those features recommended in the preliminary architecture document that were amenable to quantitative analysis by means of a model, were examined for effectiveness in the application domain of the architecture. Proper weights were given to each application. A number of surprising results were

obtained thereby. These in turn led to the elimination of certain features, the increase in others, and the incorporation of totally new features not previously considered.

In all, the modeling effort provided a second round of more sensitive feedback to the entire architectural task.

(14) <u>CCC Architecture</u>. The net result of the entire previous effort was to create a body of documentation, attitudes, knowledge, observations, etc., which were incorporated in the architecture as presented here. The process by which the progression from steps (11, 12, 13) to step 14 was made - the architecture itself, is outlined in the discussion of the next section.

3.2.2 Highlights of Major Architectural Decisions

3.2.2.1 General

Figure 3.2-2 is a tree of the major decisions which led to the proposed architecture. The discussion will proceed from the top to the bottom of the tree, one node at a time. Attention will be focused primarily on the rejected alternatives. Many lower level decisions are not shown. The reasoning behind these is given in detail in Volume IV of this report.

3.2.2.2 Monolithic vs. Multi-Computers

3.2.2.2.1 General

The first major decision was to decide between a monolithic computer approach or a multi-computer approach. A monolithic computer approach is one in which only one computer does the entire job. There is, in these applications, an identical standby computer which is relegated to low priority and off-line tasks. The monolithic approaches primary virtue is that of conceptual simplicity. The monolithic approach can be further subdivided into three subclasses:

(1) A single identical computer used for all applications.

(2) A uniform family of upward compatible computers, having identical repertoires, but differing only as to speed and other resources.

(3) A nested family of computers in which the smaller members are a logical subset (i.e., their repertoire is a subset) of the larger members.

ALL SYSTEMS

Monolithic

Uniform
Family
X

Nested
Family
X

Single
Computer
X

Multi-Computer

Uniform
Family
X

Nested
Family
X

Single
Computer

Micro
X

Mini
X

Midi

Maxi
X

Super
X

Array
X

Full Interconnect

Private
Memory
X

Shared
Memory

Private &
Shared Memory
X

Duplex
Unibus
X

Duplex
Multibus
X

General
Matrix

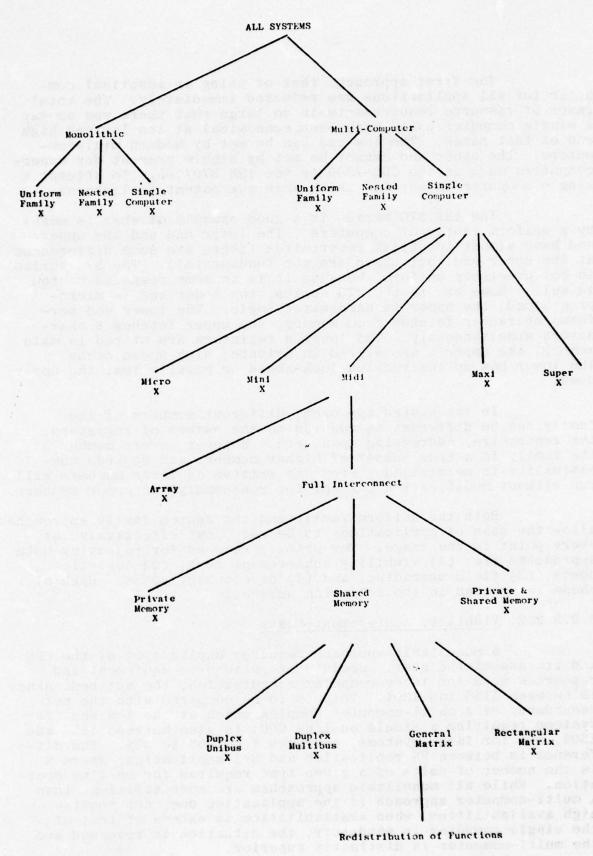Rectangular
Matrix
X

Redistribution of Functions

Fig. 3.2-2 - Architecture Decision Tree

II-121

The first approach, that of using an identical computer for all applications was rejected immediately. The total range of resource requirements is so large that there was no way a single computer could have been economical at the low and high end of that range. The low end can be met by modern mini-computers. The upper end cannot be met by single present day super-computers such as the CDC-7600 or the IBM 370/195. To attempt a single computer to service this range was patent foolishness.

The IBM 370 series is a good example of what is meant by a uniform family of computers. The lower end and the upper end have almost identical repertoires (there are some differences at the upper end, but these are not fundamental). The 370 series is not precisely uniform, because it is in some respects nested as well. However, in the 370 series, the lower end is micro-programmed, the upper is hard-wired logic. The lower end performs character fetches from memory, the upper fetches 8 characters simultaneously. The lower's registers are stored in main memory, the upper's are stored in private, high speed cache. The lower has no instruction look-ahead or parallelism, the upper does.

In the nested approach, different members of the family can be different as regards to the number of registers, the repertoire, addressing span, etc. However, every member of the family is a true subset of higher members and upwards compatibility is maintained. Programs written on lower members will run without modification (except for reassembly) on upper members.

Both the uniform family and the nested family approaches allow the span of applications to be met, cost effectively, at every point in the range. The primary reasons for rejecting both approaches are: (1) viability achievement cost, (2) logistics costs, (3) field upgrading, and (4) development costs. Each of these is treated in the following subsections.

3.2.2.2.2 Viability Achievement Costs

A monolithic approach requires duplication of the CPU and its associated gear. Because of switchover equipment and resources used for inter-computer coordination, the net redundancy is between 115% and 130%. This is to be compared with the net redundancy of a multi-computer complex which at the low end (for systems requiring a single on-line CPU) is also between 115% and 130%, but for large systems, averages from 20% to 30%. The difference is between 2N replication and N+1 replication, where N is the number of units of a given type required for on-line operation. While all monolithic approaches are more efficient than a multi-computer approach if the application does not require high availabilities, when availabilities in excess of that of the single computer is mandatory, the situation is reversed and the multi-computer is distinctly superior.

II-122

3.2.2.2.3 Logistics Costs

In either monolithic approaches, nested or uniform, when the entire series is taken into account, there are more chip types, card types, racks, backplanes, power supplies, etc., than in an equivalent architecture based on multi-computers. This means a more varied spares pool, longer training for technicians, a more complex spares procurement, etc.

3.2.2.2.4 Field Upgrading

It is virtually impossible to upgrade a monolithic system in the field without interruption of service. If the hardware and software architectures have been properly done for the multi-computer, the small installation is designed as a degraded version of the larger installation. For example, assume that the largest configuration of a particular kind of switch required five on-line computers, and that the smallest required two on-line computers (this is the case with overseas AUTODIN). The basic software is written around a five computer configuration. The two computer configuration operates as a five computer configuration in which three computers have "failed". Field upgrading is done by installing another computer and "repairing" it. This can be done without interrupting on-line operation.

No such neat approach is possible with a monolithic computer architecture. At best, the standby computer must be removed and replaced with the next one up the line. During the hardware checkout of the new standby, the system is vulnerable to a CPU failure. Once the higher level standby is checked out, it can be relegated to the on-line tasks and replacement of the other computer can be done. The system is again vulnerable, until the entire job is finished; possibly a period of several months. Furthermore, there is the logistics problem of what to do with the replaced units. It is not clear that there will be an installation for them. In the multi-computer architecture, there are no replaced units to ship to some warehouse or other installation - they remain part of the configuration.

3.2.2.2.5 Development Costs

The development costs are obviously higher for a monolithic approach since there are more computers to be designed, developed, checked out, etc. Of the two, the nested family is probably slightly less expensive.

3.2.2.2.6 Summary

The monolithic computer approach was rejected (and therefore multi-computers accepted) because:

(1) The range of applications could not be met with a single main-frame economically; only a family could be cost effective.

(2) The monolithic approach was decidedly inferior with respect to:

    (a) achieving the desired system availability;

    (b) logistics and life cycle costs;

    (c) ease and cost of field upgrading, and;

    (d) total system development costs.

3.2.2.3 Multi-Computers - Family or Single

3.2.2.3.1 General

Having decided to use a multi-computer approach, it does not follow that the multi-computer complex is to be constructed out of a single computer type. Again, the same decision regarding the use of a single CPU design, a nested family, or a uniform family must be made. Since the typical CPU of a multi-computer architecture tends to be relatively small when compared to the upper end of a monolithic family, the range in CPU size would not be so great. The decision was made to use a single computer design. However, some aspects of a nested design may still emerge.

3.2.2.3.2 Family Approaches

The disadvantages of a family approach, discussed in Section 3.2.2.2 above, still hold, but to a lesser degree. For example: in a multi-computer family approach, a large CPU might be used for the functional control of a message switch, to do message processing and other common functions, and a smaller CPU for front-end functions. This is essentially the approach taken in the TRI-TAC message switch (although the CPU's are not of the same family). The small CPU could be used to perform front-end functions, disc control, small circuit switches, and digital concentrators. If a nested approach were chosen, the bottom of the series could be a micro-computer which would be used in many other roles as a controller, as a micro-computer within the CPU, etc.

Once the decision has been made to go the multi-computer route, justifying the use of a single computer type is not the main concern, but rather justifying more than one type. The principal advantage of a family within a multi-computer architecture is to achieve a closer match of resources to needs, principally to reduce the degree of over-design. The family approaches

were rejected for the following reasons:

    (1) The savings due to the reduction in over-design would not have been significant.

    (2) Some over-design is desirable in all applications to minimize field upgrading.

    (3) The disadvantages of a family, viz: viability achievement costs, logistics costs, field upgrading costs, and development costs, still remain although in a reduced form.

3.2.2.3.3 Role of Micro-Computers

It is expected that one or more micro-computers, incompatible from the point of view of repertoire and other family characteristics will be used throughout the various units of the architecture. This, however, is a logic design decision, effectively an implementation detail. No attempt will be made to have the micro-computers be in any way part of a series which includes the CPU itself.

3.2.2.4 Computer Size - Micro, Mini, Midi, or Maxi

While exact measures of computer sizes have yet to be devised, the following characteristics are generally agreed upon:

    (1) **Micro-Computer**

        4 to 8 bit word length; repertoire of 16 to 64 instructions; addressing span of 16K characters; two or three specialized registers; no indirect addressing or indexing; one interrupt line; one priority level; no DMA; no explicit channel.

    (2) **Mini-Computer**

        16-bit word length; repertoire of 32 to 128 instructions; addressing span of 132K characters; 8 or 16 generalized registers; indirect addressing; indexing; multiple interrupt lines; two priority levels; DMA; simple channels.

    (3) **Midi-Computer**

        32-bit word length; repertoire of 128 or more instructions; addressing span of 132K to 4 million characters; 8 or 16 generalized registers; full addressing facilities; multiple interrupt lines; multiple priority levels; fully developed I/O structure.

(4)   Maxi-Computer

> 32 to 64 bit word length; repertoire of 256 or more
> instructions; addressing span of 4 million or more
> characters; several sets of generalized registers;
> floating point hardware; fully developed I/O struc-
> ture; instruction look-ahead; parallel ALU's;
> virtual memory; fully developed memory protection.

There was never any real question regarding the selec-
tion of either a micro or a maxi. The micro-computer would have
consumed more resources in coordination and communication with
the rest of the complex than would have been gained by adding it.
The overhead would have been unbearable and not effective. The
maxi was out of the question, although many features of such com-
puters have been incorporated. This left the choice between a
mini or a midi: with the primary issue being one of word length
and address span. Comparisons of throughput and program space
requirements for message switches using 16-bit and 32-bit machines
showed that the 16-bit machine had less than 1/3 the throughput
of a comparable 32-bit machine. In other words, the short word
length exacted a disproportionate penalty which was felt both in
the increase in program space (not only on an instruction basis
but on a total character basis) and in decreased throughput.
The recommended CPU, in many respects corresponds to the upper
end of a mini line, but has features traditionally found only in
large computers. It is best characterized as a midi.

3.2.2.5 Parallel Array vs. Full Interconnect Architecture

A parallel array architecture, of which ILLIAC IV is
the archtype, is characterized by a limited form of inter-com-
puter connections. In a two dimensional array (ILLIAC IV again)
each processor can connect only with its nearest neighbors. This
scheme can be extended to three or more topological dimensions.
The crucial issue is whether full interconnectability is provided
or limited interconnectability. If full interconnectability,
meaning the ability of any one CPU to fulfill any functional
role, is required (as it is for N+1 replication) then full inter-
connectability follows as a corollary.

A second characteristic of many array computer architec-
tures, again typified by ILLIAC IV, is simultaneous processing
in parallel by many computers, executing essentially the same
instruction stream. This is clearly not effective in any of the
applications of the CCC. An extremely large toll office, with
say 40,000 trunks, might be processing 100 to 200 calls per
second. At any instant of time only a few of these would be in
the same state, permitting truly parallel processing. In message
switching, while there may be several messages in progress simul-
taenously, because of the great diversity of mode, code, formats,
etc., it would be unlikely that any two messages would be in the

II-126

same state.

It is for the above reasons that a full interconnect architecture in which every CPU was identical was chosen, capable of fulfilling any functional role, and within that role, operates quasi-independently of any other computer in the complex (i.e., working on its own instruction stream).

3.2.2.6 Private vs. Shared Memory

In a private memory system, every CPU controls its own memory modules, which are inaccessible to other CPU's in the complex. In a shared memory system, all memories are accessible by all CPU's. The advantages of private memory are:

(1) Simpler software organization.

(2) Reduction in the memory switching hardware.

(3) Prevention of memory poisoning.

(4) Net higher memory speed.

This latter comes about because the CPU and memory are closely coupled. The main disadvantages of private memories are:

(1) Less graceful degradation.

(2) A choice between reduced interchangeability or memory wastage (since, if every CPU is to be interchangeable they must all have the memory required by that CPU which functionally needs the largest amount of memory.

(3) Increased processing time for memory-to-memory moves.

(4) Larger program space, buffer space, and table space.

Shared memory has the following major advantages:

(1) Total reduction in memory for a given availability.

(2) Little or no wastage.

(3) Full interchangeability.

(4) Reduced processing by elimination of inter-computer chit-chat.

Shared memory has the main disadvantage of:

(1) Complex memory interchange/interconnect network.

(2) Possibility of poisoning.

Of these disadvantages, memory poisoning was the more serious. However, if proper memory protection features are provided, memory poisoning is not a problem. The larger interconnection network is not that much larger, and the reduction in replication and memory wastage compensates for this increase several times over.

A compromise between purely private and purely shared was investigated - this would have used a private-plus-shared memory system. This approach, upon closer examination, still had many of the disadvantages of both approaches, and since the private portion tended to be smaller than typical memory module sizes, the savings were not all that great. Eventually, most of the functions which would have gone into a private store ended up in a very small, private cache store in every unit, not just in CPU's.

3.2.2.7 Interconnection Topology

3.2.2.7.1 General

The interconnection topology of the CCC probably represents the most important set of architectural decisions made. These are discussed in detail in Volume IV, however their importance warrants summary here.

Consider the following argument:

- If N+1 replication in the interest of achieving a given availability is to be used, then every unit of a given type must be completely interchangeable with all other units of the same type.

- If a unit is to be interchangeable with other units of the same type, then every unit of that type must be configured with the resources required in the worst case.

- If units are not configured thusly, a special spare unit with identical resources must be provided; in which case duplication (2N redundancy) rather than N+1 replication is provided.

- If units are to be interchangeable, then they must be capable of being interconnected in accordance with the worst case pattern.

● If units are to be interchangeable, then they must be switchable into any functional configuration.

Note that the requirements of interchangeability in the interest of economical viability imposes internal switching requirements that are over and above those that would be needed in a simplex commercial system.

Consider the following statements:

● Every computer complex has a switching network used to connect CPU's to memories.

● Every computer complex has switching networks used to connect channels to memories and channels to computers.

● There are other, smaller, switching networks used for various purposes within the complex.

● Full interchangeability of units implies that the above switching networks are permutation networks – i.e., capable of achieving any permutation of inlets against outlets.

● Traffic in this network can be defined in terms of erlang loads of interconnections, in the classical manner of circuit switching matrix design.

● Except for speed, then, the design of the internal interconnection topology of a computer complex is a classical traffic engineering problem, using methodologies that date back to the turn of the century.

3.2.2.7.2 Bus Structured Topologies

In the light of the above observations, a justification of a properly engineered interconnection network should not be required, but rather a justification for a bus structure, had one been chosen. A bus is a particularly simple, highly specialized case of a switching network - corresponding to a primitive party line. If availability had not been an issue, then the rich, permutible interconnection structure would not have been required. The total number of possible interconnection permutations would have been small. Bus duplication would not have been needed, and bus control replication would have been unnecessary. Had this been the case, the use of several, functionally organized, unreplicated, individual busses, as the most cost effective means of achieving the required internal interconnections, would have been recommended. However, viability is a central issue and a

means must be provided for switching the busses and the bus controls. The busses, because of the large number of units (compared to commercial systems and particularly monolithic architectures) that must be terminated in the complex, are a significant part of the system cost and contribute noticeably to the failure rate. Provision of bus switching merely pushes the viability issue off to that of the bus switch, which then becomes the limiting factor. Switching the bus switch pushes it off further - continuing in this manner, it is found that total system availability is going down rather than up. A different structure, with inherent graceful degradation is essential.

3.2.2.7.3 Rectangular Matrix Structures

A single stage rectangular matrix such as is commonly used between memories and the rest of the complex, typically buried in the memory control logic, except for viability issues, is cost effective when the number of ports is small. The rectangular matrix follows an N^2 growth law. The single stage rectangular matrix does not have inherent graceful degradation and is therefore as vulnerable as the bus. Again, had viability not been in issue, the choice would have been between single stage matrices and busses - which is to say a choice between time division and space division.

3.2.2.7.4 Multi-Stage Matrix Topology

The multi-stage telephone switching matrix is the general case of which the bus and the single stage rectangular matrix are special cases. The multi-stage matrix follows a NlogN growth law rather than an N^2 law and is therefore more cost effective than the single stage matrix. While the bus has the superficial appearance of a linear growth law, this is deceptive since speed is not free. That is, to handle twice tne traffic, the bus must be twice as fast. Generally, growth laws for time division have been closer to N^2 than to N, because of the speed requirement.

The multi-stage matrix has the following advantages:

- NlogN growth law.

- Modular growth.

- Possibility of exact sizing to meet internal traffic needs.

- Total interconnectability; total interchangeability.

- Inherent graceful degradation without duplication.

- N+1 redundancy rather than 2N.

- Capable of field expansion without service interruption.

In other words, it has all the characteristics that have been sought in all other aspects of the architecture. Being a more general case, it may in some installations be reduced to a simple bus or to some simple, single stage rectangular matrices.

3.2.2.8 Redistribution of Functions

3.2.2.8.1 General

One of the characteristics of the architecture is the redistribution of functions as compared to the way those functions have been implemented in the past, especially in monolithic architectures. By "redistribution" is meant that the partition of functions into units does not follow the typical. This has come about because of economic and viability considerations. Here it is not a question of choices between alternatives, but rather reviewing the choices made and why.

3.2.2.8.2 Separation of I/O Channels and CPU

In classical architectures, I/O channels have typically been associated - that is, closely coupled, to the CPU or CPU's. In this architecture, the channel emerges as a totally separate, independent unit. LSI permitted the elimination of diverse channel types, allowing the same high speed, cycle stealing channel to be used for all devices. Communication systems, particularly message switches, have far more channels than is typical in commercial systems. Had channels been kept closely coupled to the CPU, then every CPU would have had to have the number of channels required by the worst case CPU in the complex. This would have doubled or tripled the number of channel units, and easily doubled the switching matrix required to connect devices to the channels. The present separation of channel and CPU eliminates this device-channel switch altogether, providing thereby a far more economical approach.

3.2.2.8.3 Separation of Interrupt Functions and CPU

Systems have gradually been moving towards a programmable interrupt structure and away from a hard-wired structure. A programmable structure means that each functional CPU can be configured with the interrupt hierarchy best suited to it rather than the interrupt hierarchy required for the worst case. The dissociation of channels and CPU means that the overwhelming majority of interrupts did not automatically come into a specific CPU. Given an I/O control structure that incorporates command

chaining, data chaining and I/O termination chaining, a detailed
examination of the remaining interrupts shows that timely re-
sponse is not statistically required if relative order among
interrupts is maintained. Thus, it is feasible to think in terms
of interrupt reactions measured in milliseconds rather than micro-
seconds. Consequently, there is no need to have every CPU con-
tain the entire sophisticated interrupt mechanism. All interrupt
functions for the entire complex can be centralized in a smart
interrupt handler (the ICU). This was done in the CCC architec-
ture. Therefore, instead of having the same worst case interrupt
mechanism in every CPU, the interrupt hardware was placed in one
unit (and its standby), the ICU, and the individual CPU's were
left with a rudimentary, non-programmable, identical interrupt
structure. The final phase of the interrupt evolution was the
realization that the combined functions of the ICU were suffi-
ciently diverse and complex to implement them as another stored
program CPU rather than as a specialized ICU.

3.2.2.8.4 Migration and Expansion of Protection Features

The typical large CPU has memory protection hardware
which prevents the inadvertent poisoning of memory due to faulty
code or malfunctions. That hardware is typically in the CPU's
themselves. In the case of the CCC, that would have meant that
each CPU had to have the entire protection tables for the entire
memory. Furthermore, the need for reconfiguration of all units
in the interest of viability meant that physical identification
could not be used for memory protection, but only logical identi-
fication. This meant that every CPU would have had to carry a
map of the relation between the logical and physical identities
of every memory unit. Unlike commercial systems in which memory
boundary violations are dominated by bugs in application programs,
communication system memory boundary violations occur because
of malfunctions, in which channels and devices are to blame as
much or more often than CPU's. Consequently, not only must pro-
tection be provided against CPU failures, but against device
failures, ICU failures, etc. Taking all of this into account,
it is far more economical to place memory protection functions
in the memories themselves rather than in the CPU.

The second step was to expand the protection concept
to provide command protection. This is essentially an expansion
of the "privileged", or "executive" mode instructions in commer-
cial machines, to four categories of instructions. The third
step was to realize that all units must be protected against mal-
functions in any unit that they communicate with. The CCC not
only provides the typical memory protection, but provides com-
plete inter-unit protection for both accesses and command types.

3.2.2.8.5 Redistribution and Expansion of I/O Functions

The separation of channels from the CPU was tantamount

II-132

to the separation of I/O hardware from the CPU. All I/O hardware
has essentially been redistributed into the channel and the ICU
(which is another CPU). As the architecture emerged, different
unit types evolved - channels, CPU's, memories, ICU, clocks, sub-
matrices, etc. All units had to be controllable. All units
(since they all contained cache memories for protection and pri-
ority) had to have the ability to modify or reload their private
stores - especially since the contents of those cache memories
would vary with the functional assignment of the unit - again
the interchangeability issue. Protocols for inter-unit communi-
cations had to be devised. Protocols and instructions for unit
control and logical configuration had to be devised. What emerged
was the recognition that there was another repertoire - the inter-
unit or IU repertoire, of which the I/O repertoire was a subset.
That repertoire was subdivided into two broad classes of instruc-
tions - unit generic and unit specific instructions. Unit generic
instructions apply to all units of whatever type, unit specific
apply to individual unit types. A further subdivision of the IU
repertoire into micro and macro instructions was made. These
two classes roughly correspond to those instructions that are
issued by hardware for inter-unit protocols transparent to the
programmer (IU micros) and those which the programmer was likely
to issue directly for I/O operations, for configuration control,
or for diagnostic purposes (IU macros).

PREFACE

Final Report, Volume III describes the analytical modeling effort which was undertaken to test various architectural features and to provide answers to specific processor architectural and feature questions.

The modeling tool, GASM, was developed by Data Systems Analysts, Inc., under the direction of Dr. Boris Beizer.

This volume contains only a general description of GASM, and how it was used. For a detailed description, the reader is referred to the user's manual, "General Analyzer for System Models, GASM-II", IM-5088-2, dated September 4, 1975, which is available from RADC or Data Systems Analysts, Inc., North Park Dirve, Pennsauken, N.J.

The information contained in this volume was assembled and written by Dr. B. Beizer and Mr. K. Hagstrom.

VOLUME III

MODELING

TABLE OF CONTENTS

TABLE OF CONTENTS - (Continued)

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ABBREVIATION	TITLE
ACC	Accountability
ACK	Acknowledge
ACP-127	Standard AUTODIN Data Fromat (one of two standards)
AMME	Automated Multi-Media Exchange
AVG	Average
BDS	Base Distribution Switch
BH	Busy Hour
CCC	Communications Computer Complex
CHAR	Character
CKT	Circuit
CPS	Communications Processor System
CPU	Communications Processor Unit
C/S	Circuit Switch
FE	*Front End*
GASM	Generalized Analytical Systems Model
HR	Historical Accountability
HSL	High Speed Line
HT	Holding Time
I/O	Input/Output
ITA #2	Data Format Code Name
ITS	In-Transit Storage
JANAP-128	Standard AUTODIN Data Format (one of two standards)
LMF	Language Media Format
LSL	Low Speed Line
MP	Message Processor/ing
MS	Message Switch
MSG	Message
MSL	Medium Speed Line
NACK	Negative Acknowledge
N/D	Narrative Data

ABBREVIATIONS - (Continued)

ABBREVIATION	TITLE
P/S	Packet Switch
RI	Routing Indicator
SF	Store and Forward
VHD	Very High Speed Dedicated Trunk
VHL	Very High Speed Line

VOLUME III

MODELING

1.0 INTRODUCTION

The purpose of the modeling effort was to test the cost
effectiveness of various computer concepts with regard to per-
forming the communications switching tasks of the foreseeable
future. For example, "Should the system have a clear memory in-
struction?" is typical of the kinds of questions that were
answered with the modeling effort. In principle, the approach
was simple. Two models were generated, one which had the clear
memory instruction, and one which did not. Then the performance
of the two resulting systems was compared to see if there was a
payoff. That comparison could be done either from the point of
view of processing delay improvement at a given cost and traffic
throughput; throughput improvement at a given cost and delay; or
cost improvement at a given throughput and delay. The specifics
of the comparison were not important - what was important was
that a comparative measure was derived, rather than an absolute
measure. While the assumptions embodied in the model may have
caused errors on an absolute basis, it did prove to be a sensi-
tive tool for comparison purposes which was the reason for which
the model was created.

Subsequent sections discuss the model software, the method
used to simulate the existence or non-existence of various fea-
tures in the model, the specific design options which were
examined, and the results of the model runs.

2.0 DESCRIPTION OF MODEL SOFTWARE

The model software consists of the Master Model files and a
Generalized Analytical System Model (GASM).

Software design data in the form of flowcharts, and to a
lesser extent, hardware and traffic characteristics were used to
prepare source math models for the various software components
of the systems being modeled. These mathematical models were
then manipulated to produce reduced math models. These reduced
math models constitute the Master model.

2.1 Model Files

GASM is a program for a Generalized Analytical System Model
for message and circuit switching systems. The purpose of the
model is to allow the rapid assessment of the effect on the per-
formance of a computer controlled communication system of changes
in hardware, software, traffic and other conditions. The model

consists of a system of interrelated programs written for a time-sharing system. It was the main working tool used for the assessment of circuit and message switching software and hardware under the CPS project.

Figure III-1 shows the relation between GASM and the Master Model files and the means by which both are used in support of the modeling effort on the CPS study.
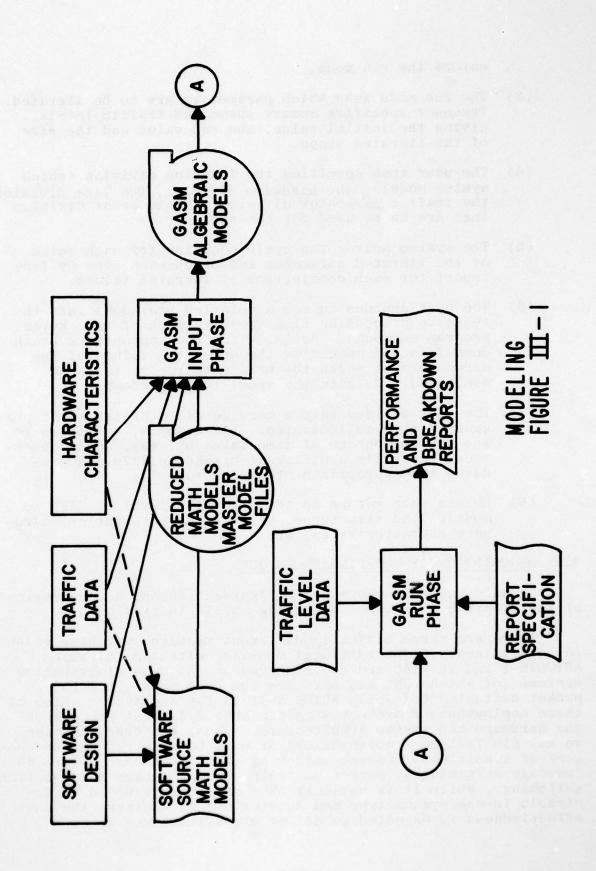
The GASM input phase allows the user to specify functions and values for hardware, software and traffic characteristics. Six sets of specifications can be provided: FUNCTION, HARDWARE, LINE, TRAFFIC, PRINT, and BREAKDOWN REPORT, in addition to allowing the specification of a number of other parameters on-line. Each set of specifications is stored as a file and can be used in arbitrary combination with other sets of specifications. For example, a given set of traffic characteristics can be examined with a number of different hardware characteristics, or a given set of hardware characteristics with a number of different traffic characteristics.

Once entries have been provided for each of the above six model divisions, the user may enter the run mode in which he specifies traffic level data, the kind of output he wants, and the particular files that are to be used to calculate the performance parameters of interest. GASM solves the cycle equation for the system and calculates all basic unknowns and then goes on to print one or more reports as specified. Typical output parameters include: message delays for each of several kinds of traffic, core utilization, channel utilization, running time of various routines, etc. Since GASM is completely flexible, the user has the option of defining whatever kind of output reports or functions he wants. At the conclusion of a given report, additional runs can be specified as needed.

2.2 Operational Characteristics

The following scenario describes the steps that might be followed in examining a system's characteristics using GASM. It will be assumed that a basic set of Master Model files has been defined in the past and that the user has a choice of one or more of these.

(1) The user initiates a run and is asked which mode of operation he wishes to use. Assume that he wants to change the memory speed and determine the influence of channel transfer rate on messages as a function of message switched traffic.

(2) The user specifies that he wishes to create a new hardware parameter file. He calls in the existing hardware parameter file, modifies it and renames it. He then

MODELING
FIGURE III-I

NORTH ELECTRIC CO   GALION OHIO
COMMUNICATIONS PROCESSOR SYSTEM.(U)
JAN 77   K HAGSTROM, B BEIZER

F/G 17/2

F30602-73-C-0314
RADC-TR-76-394-VOL-1/2/3   NL

END
DATE
FILMED
4-77

enters the run mode.

(3) The run mode asks which parameters are to be iterated. The user specifies memory speed and traffic levels, giving the initial value, the end value and the size of the iterated steps.

(4) The user then specifies the function division (which system model), the hardware division, the line division, the traffic parameter division and the print division that are to be used for the run.

(5) The system solves the cycle equation for each value of the iterated parameter and produces a line by line report for each combination of iterated values.

(6) The user decides to see a detailed breakdown into the relative processing time spent in each of say, seven, program segments. He calls for the appropriate break-down file and specifies the numerical values of the parameters at which the breakdown report is to be evaluated. He gets the specified breakdown.

(7) The user decides that a more detailed breakdown of processing time is indicated. For example, he wishes to see the percentage of time taken by, say, code conversion. He calls a different breakdown file and produces the appropriate breakdown report.

(8) He can then return to the executive level of GASM to modify line structures, traffic characteristics, hardware characteristics, etc.

3.0 DESCRIPTION OF MODELING TECHNIQUE

As an example of the modeling approach taken, a description of the creation of message switching models is presented.

There are three basically different message switching roles to be examined: (1) traditional message switching (of which AUTODIN-I and TRI-TAC are the archtypes), (2) base distribution systems (of which LDMX and AMME are the archtypes), and (3) packet switching (of which AUTODIN-II is the archtype). Each of these deployments represent significantly different demands on the hardware and system architecture. Thus, whatever question we ask (Is feature X worthwhile?) it must be answered in the context of traditional message switching (hereafter referred to as "message switching"), packet switching, and the base distribution switching. While it is unlikely that any feature would be desirable in one system type and deleterious in another, the cost effectiveness is expected to differ significantly. Given a
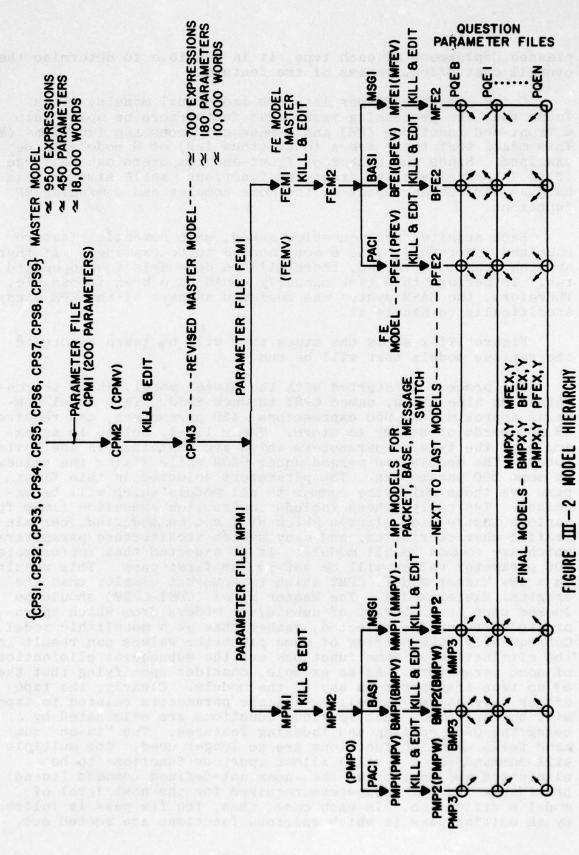
planned deployment of each type, it is possible to determine the overall cost effectiveness of the feature.

Going somewhat deeper into the individual models, it is found that the processing tasks must furthermore be split into a front-end functions (FE) and a message processing functions (MP). This means that there are a total (thus far) of 6 models to be examined. Since the number of front-end computers can be large (3 to 4 is typical) the front-end functions, while simpler, can have as serious an impact as the more complex and demanding MP functions.

Each architectural question asked, each potential feature examined, therefore, adds 6 more models to be examined. If there are ten questions to ask, there will be 60 models developed and run. To perform this task manually would have been impossible. Therefore, the GASM system was designed as part of the CPS study specifically to handle it.

Figure III-2 shows the steps that will be taken to create the various models that will be run.

The process is started with the master model, which is contained on nine files, named CPS1 through CPS9. This model contains approximately 950 expressions, 450 parameters, and requires 18,000 words of memory to store. Table III-1, below, is an example of the type of parameters which are contained in the Master Model. The models are merged under GASM while fixing the values on some 200 parameters. The parameters selected in this first pass are those which are common to all models which will be examined. Typically, these include instruction execution times for various instruction classes which will not be modified, certain traffic characteristics, and many system architecture parameters which are common to all models. It is expected that approximately 200 parameter values will be set in the first pass. This results in a new Master Model, CPM2 which is somewhat simpler than the original Master Model. The Master Model (CPS1-CPS9) should be looked upon as a library of subsidiary models from which appropriate parts can be selected, rather than as a monolithic model. Consequently, the setting of some parameter values can result in the elimination of some functions and the subsequent elimination of some parameters. As an example, consider specifying that there be no tape transports in any of the models. Clearly, the tape driver subroutine, as well as all the parameters related to tapes will be obviated. Such spurious functions are eliminated by using the GASM editing and indexing features. The "in-nc" command tells us which functions are no longer used. The multiple kill command (al f - kill) allows spurious functions to be eliminated wholesale, and the index not-defined command (in-nd) provides a list of parameters required for the next level of model modification. In each case, then, the fix pass is followed by an editing pass in which spurious functions are rooted out.

FIGURE III-2 MODEL HIERARCHY

TABLE III-1

TYPICAL EXAMPLES OF M/S PARAMETERS

Description	Unit	Value
Main memory speed - for one memory cycle	usec/cycle	1-
Number of memory cycles for normal instruction (e.g., load, store)	cycles	2-
Number of memory cycles for indexed normal instruction	cycles	2-
Number of memory cycles for single level indirect instruction	cycles	3-
Number of memory cycles for indexed indirect instruction	cycles	3-
Number of memory cycles for base of long instruction (e.g., shift)	cycles	1-
Number of memory cycles for literal mode instruction	cycles	1-
Fraction of a memory cycle for single bit of shift	cycles	.05
Number of memory cycles for special instruction (e.g., extract-insert-store)	cycles	3-
Number of memory cycles for indexed special instruction	cycles	4-
Number of memory cycles for "short" I/O instruction	cycles	2-
Number of memory cycles for base of indexed shifts	cycles	3-
Does the CPU have a clear memory instruction?	logical	Yes (1)
Does the CPU have a ledgering instruction?	logical	No (0)
Does the CPU have a data move instruction?	logical	TBT
Does the CPU have a block move instruction?	logical	TBT
Does the CPU have a link/unlink instruction?	logical	TBT
Does the CPU have a chained I/O?	logical	Yes (1)
Does the CPU have a character address mode?	logical	Yes (0)
Does the CPU have a bank I/O limitations (i.e., I/O only from specified banks)?	logical	No (1)
Does the CPU have a hardware index reg. storage for each priority level?	logical	TBT
Is the CPU paged?	logical	TBT
Does the CPU have complete scatter/gather I/O?	logical	Yes (1)

<u>TABLE III-1</u> (Continued)

Description	Unit	Value
Number of memory cycles for program level change + return	cycles	20-30
Number of memory cycles for raw interrupt overhead	cycles	20-30
Exactly one character per word (character memory)	logical	No (0)-
Number of characters per word	number	4-

NETWORK PARAMETERS

Description	Unit	Value
Is there an interface to a CKT switch?	logical	Yes (0)
Are the CKT switched lines encrypted?	logical	Yes (0)
Is there packet switched traffic?	logical	Yes (0)
Probability that a CKT switched line is encrypted	prob.	.8
Percentage of CKT switched lines that are "permanently" assigned	prob.	
Total number of lines terminated at switch	number	50-400
Total number of Mode I lines terminated at switch	number	50-150
Total number of packet switched lines at switch	number	5-15
Average line speed	CPS	Variable
Number of switches connected to this one	number	6
Number of switches in entire network	number	20
Number of circuit switched trunks	number	Variable
Number of CKT switched trunk groups	number	10
Number of RI's in RI table	number	2000
Number of destinations in destination table	number	3000

* TBT = To Be Tested

The largest use of this comes about in generating the front-end Master Model (FEM1) in which most of the MP functions are eliminated. It is also heavily used to eliminate the front-end functions from the MP model, and in doing the packet, message, base model split.

The creation of the CPM2 model from the CPS1-CPS9 models results in simplifications. Many of these simplifications come about because functions which were algebraic become numerical. The values of such functions are retained in a file for future reference. In the case of CPM2, the file is called CPMV.

The CPM2 model is revised by editing and killing of spurious functions to create the CPM3 model. The model selection switches are then set, along with two sets of parameter values, corresponding to all front-end models and all message processor models respectively. This fix pass (two different fixes) results in the common MP model (MPM1) and the common front-end model (FEM1), along with the numerical value files MPMV and FEMV respectively. MPM1 and FEM1 are then simplified by killing spurious functions to create MPM2 and FEM2.

The next phase in the model creation procedure is to perform a three-way split of the MP and FE models. This is done in two passes. Six parameter files are used for this purpose - PAC1, BAS1, MSG1, PAC2, BAS2, and MSG2, which contain the packet switching, base distribution switching, and message switching parameter values respectively. The same parameter file is used for the MP and FE models. This can be done safely since any parameter not used in the fix will be ignored by GASM. Six models result from the first pass: PMP1, BMP1, MMP1, PFE1, BFE1, and MFE1 corresponding to the Packet, Base, Message, MP or FE models as appropriate. Similarly, numerical value files are created as a result of the fix: PMPV, BMPV, MMPV, PFEV, BFEV, MFEV. A second pass creates models PMP2, BMP2, MMP2, PFE2, BFE2, MFE2 and parameter files PMPW, BMPW, MMPW, PFEW, BFEW, and MFEW. Killing and editing are applied to simplify the models, resulting in the final set of six models: PMP3, BMP3, MMP3, PFE3, BFE3, and MFE3.

The final step is to set switches and parameter values corresponding to the question being asked. A base value parameter file PQEW is used across all six models to create the base version models against which the various options will be compared. A separate parameter file which sets all remaining parameter values (except traffic level) is used for each design question. Application of this parameter file (say PQEX) to all six models results in six models which are specialized to the option being investigated. Traffic varies over a range from no traffic to the point at which the system hypothetically breaks down under overload. The resulting curves can then be compared to determine the cost effectiveness of the option.

4.0 MODELING PARAMETERS

For each run performed with GASM, a set of values (points) is produced which, when plotted, will result in a curve similar to that shown in Figure III-3. The curve shows the processing delay versus traffic units. The scales shown on the coordinates may vary from system to system and the curves may show steeper knees but the general shape of the curves produced will be similar. The traffic units may be calls/sec., messages/sec., etc.

Note in the figure that the Average Busy Hour Maximum has been marked. This point is the traffic versus delay which is usually specified and which any viable system must be capable of handling.
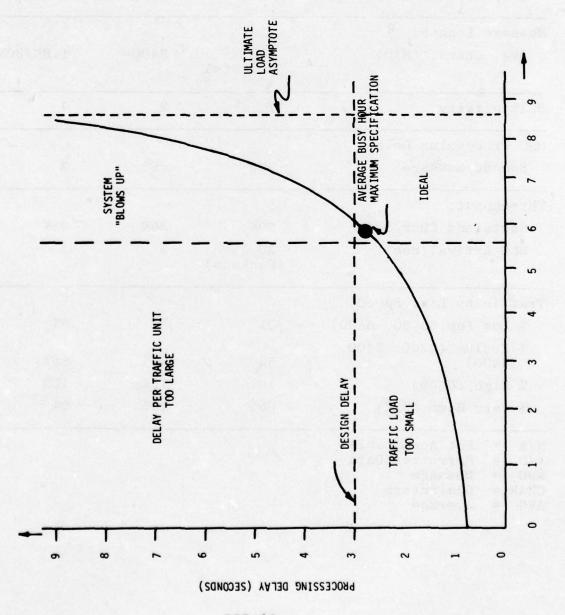
It is not enough, however, to know only the location of this point but to know what the entire curve looks like. That is, "how sharp is the knee", "what percent increase in traffic causes the system to blow-up, i.e., to produce infinite processing delays".

The parameters listed below for the message, packet, circuit and base distribution switches define the Average Busy Hour Maximum used as a standard against which the various runs to test certain features and instructions were compared. In each case, a set of curves were generated which allowed the relative merits of each feature or instruction to be weighed.

4.1 Packet Switch Parameters

The Packet Switch (P/S) system is described by the characteristics of Table III-2, and the following assumptions:

(1) Three line speeds are assumed:

    a. <u>V</u>ery <u>H</u>igh speed <u>L</u>ine (VHL)

        Operates at 32,000 bps.

    b. <u>H</u>igh <u>S</u>peed <u>L</u>ine (HSL)

        Operates at 9600 bps.

    c. <u>M</u>edium <u>S</u>peed <u>L</u>ine (MSL)

        Operates at 2400 bps.

(2) Average message length = 500 characters.

(3) Average packet length = 500 characters.

(4) Header length (fixed format) = 50 characters.

III-11

TABLE III-2
TRAFFIC CHARACTERISTICS OF SYSTEMS

TRAFFIC CHARACTERISTICS	PACKET SWITCH	S/F SWITCH	BASE DISTRIBUTION
Number of Lines per Switch	40	170	380
ITS (Line Blocks)	20K	60K	120K
Message Distribution:			
% By Message (N/D)	N/A	30/70%	40/60%
Message Length:			
Avg. Chars. (N/D)	500 (Packets)	2400	1.8K/30K
Multiplicity	1	2	4
MSG Processing Delay:			
Second/Message	.06	.85-2	2
Throughput:			
Sustained Char/Sec	20K	36K	95K
MSG Arrival/Sec	20 (Packets)	5	1
Traffic by Line Speed:			
% Low (up to 600 BAUD)	0%	5%	5%
% Medium (1200, 2400, 4800)	5%	10%	85%
% High (9600)	10%	15%	10%
% Very High (32K)	85%	70%	0%

N/A = Not Applicable
N/D = Narrative/Data
MSG = Message
CHAR = Characters
AVG = Average

(5)   No multiplicity - one RI per message.

(6)   All messages use same code and formats.

(7)   Line Block length = 50 characters.

(8)   Message Distribution (BH - originating and termi-
      nating)

Percent by Line Speed	Total Messages	Traffic	Terminals
MSL 5%	7200	7E	17
HSL 10%	14400	3E	10
VHL 85%	122400	4.25E	13
TOTAL 100%	144000	14.25E	40

(9)   <u>Message Distribution</u> (by line speed)

Line Speed	Originated Messages	Terminated by Line Speed		
		VHL	HSL	MSL
VHL	61200	52020	6120	3060
HSL	7200	6120	720	360
MSL	3600	3060	360	180
TOTALS	72000	61200	7200	3600

(10) All messages are acknowledged by the receiving sta-
      tion (switch or terminal) on a packet by packet
      basis.

(11) <u>Channel Coordination</u> - a sync control character is
      received with each packet. Also, a sync control
      character is sent with each message.

(12) Average Holding Time ($\overline{HT}$) per message by line speed:

      VHL - $\overline{HT}$ = .125 sec
      HSL - $\overline{HT}$ = .75 sec
      MSL - $\overline{HT}$ = 3.50 sec

(13) <u>Accountability</u>

      Entry-Exit Accountability:

      a.   <u>Entry Traffic</u> - Maximum accountability (network)
           assumes complete responsibility for delivery of

message. Maintains messages for long period for retrieval and retransmission.

b. Exit Traffic - Historical/Recovery accounta-bility. Provides historical record of all messages which exit the network. Maintains recording of messages until delivery to receiving station is acknowledged.

c. Transit Traffic - Minimim accountability. Releases messages from intransit storage when next P/S has acknowledged receipt. Maintains delivery and channel queues. Keeps traffic statistics, i.e.,

message count

character count

statistics on acknowledgement

traffic/trunk group

trunk and line status

etc.

d. Entry and Exit Traffic - In the same switch, i.e., between locally homed terminals, is treated as Entry traffic with maximum accountability.

(14) All MSL and HSL lines are local terminals.

(15) No plain language or textual addressing.

(16) 5% of all message blocks (packets) require retransmission.

(17) Average of 6 characters per RI.

4.2 S/F Switch Parameters

The S/F switch is described by the characteristics as stipulated in Table III-2, and on the following assumptions:

(1) 24% of the terminals of the S/F are terminated on a collocated C/S. (A total of 40 terminals are terminated on the C/S).

(2) 15 terminals are permanently patched through the C/S as dedicated trunks in trunk groups to other M/S systems.

(3)   25 terminals, terminated on the C/S, are switchable to other outgoing trunk groups or to local subscribers homed on the S/F through the C/S.

(4)   All lines to the C/S operate at 32 Kb/s.

(5)   Three line types are used whose characteristics are as follows:

a.   Type 1:   Low Speed Line (LSL) (300 bps)

Operates in AUTODIN modes II, IV and V. Uses ITA #2 code in JANAP-128 or ACP-127 formats.

b.   Type 2:   Medium Speed Line (MSL) (2400 bps)

Operates in AUTODIN modes II, IV and V. Uses ASCII code and JANAP-128 or ACP-127 formats.

c.   Type 3:   High Speed (HSL) or Very High (VH) speed line (9600 and 32,000 bps)

Operates in AUTODIN modes I and III. Uses ASCII code and JANAP-128 formats.

Therefore: 2 codes, ITA #2 and ASCII, are used;
             2 formats, JANAP-128 and ACP-127 are used.

(6)   Channel Coordination

Type 1 Line:

Data Receive - recognize a steady mark (stop) signal as the idle pattern and a transition from mark to space as the start of the character frame.

Data Transmit - two methods used:

(1)   transmit data rhythmically with no external character step

(2)   transmit each character upon the receipt of a character step (release) pulse and maintain a steady mark condition upon completion of a character until the receipt of the next step pulse.

Assume that 1/2 of the Type 1 lines transmit with Method 2.

Type 2 Line:

> Same as Type 1 line as far as coordination procedures are concerned.
>
> Assume that 1/2 of the Type 2 lines transmit with Method 2.

Type 3 Line:

> Data Receive - data control characters used on a 84 character block basis. Receives idle pattern for synchronization, verification and channel checking purposes.
>
> Data Transmit - data control characters transmitted with each 84 character block. During idle periods, transmits the idle pattern.

(7) Language Media Format (LMF)

Three types of LMF's are assumed:

    (1) card format

    (2) paper tape format

    (3) magnetic tape format

Any LMF may be used with any message.

Assume an even distribution of LMF type for all line type messages.

(8) Average Holding Time ($\overline{HT}$) per message (2400 characters) is as follows for each line type:

a. Very High speed, Dedicated trunk, (VHD)

> $\overline{HT}$ = 1 second. This includes .6 seconds message transmission time and .4 seconds "dead" time.

b. Very High speed, Switchable trunk, (VHS)

> $\overline{HT}$ = 3 seconds. This includes .6 seconds message transmission time and 2.4 seconds tandem signaling time.

c. High Speed Line (HSL)

$\overline{HT}$ = 5 seconds. This includes approximately 2.5 seconds for signaling, line coordination or "dead" time.

d. Medium Speed Line (MSL)

$\overline{HT}$ = 10 seconds. This includes 2 seconds for signaling, line coordination or "dead" time.

e. Low Speed Line (LSL)

$\overline{HT}$ = 100 seconds. This includes 10 seconds for signaling, line coordination or "dead" time.

(9) Maximum traffic for 170 terminations S/F during the BH is 120E.

(10) Average of 5 message arrivals per second during the BH.

(11) Message Accountability Distribution:

Minimum Acc.	20%
Intermediate Acc.	20%
Historical/Recovery Acc.	20%
Historical-Recovery-Traffic Acc.	20%
Maximum Acc.	20%

(12) Accountability Functions:

a. Minimum Accountability

Buffer storage for active processing

Maintains delivery queues

Maintains channel queues

Releases messages (removes from intransit storage) upon receipt of acknowledge by receiving equipment

Traffic statistics are kept, i.e.,

message counts

character counts

statistics on acknowledgements (ACK's, NACK's)

b. **Intermediate Accountability** (Historical)

provides, in addition to minimum accountability facilities; historical records of predetermined processing events written to an on-line storage device. A journal of message events is kept.

c. **Historical/Recovery Accountability (H/R)**

in addition to the above, it provides recording of messages on an Intransit Storage file. Some measure of status ledgering is performed to maintain the storage structure of the messages (or blocks).

d. **Historical/Recovery/Traffic Accountability (H/R/T)**

in addition to the above, it provides recording of message blocks as received on separate recording medium. This allows retrieval and retransmission functions to be performed.

e. **Maximum Accountability**

all functions of above accountability procedures plus added redundancy and storage capabilities. Reference files are kept. Complete status ledgers redundantly stored.

(13) No plain language or textual addressing.

(14) **Message Distribution by Line Speeds:**

VHD	42.5%
VHS	27.5%
HSL	15%
MSL	10%
LSL	5%

(15) All messages are transmitted in fixed line blocks of 84 characters each.

(16) **Headers** are of two types; single RI per header in which the entire header is contained in a single line block; and multiple RI's per header in which the header is contained in 2 blocks. An average of 1/2 of input messages contain multiple header (or 2 line block headers).

(17) <u>Header Validation</u> includes validation of all fields up to the "start of routing" indicator. Rigid validation procedures used on all messages.

(18) <u>Routing Validation</u> includes checking the originating station identifier and serial number, checking each RI against the RI Table, and formatting the outgoing RI's on messages to other M/S's.

(19) <u>Block Retransmission</u> required for 5% of all trunk messages (incoming and outgoing).

4.3 Base Distribution Switch Parameters

The Base Distribution Switch (BDS) is described by the characteristics of Table III-2 and the following assumptions:

(1) Average narrative message length equals 1800 characters.

(2) Average data message length equals 30,000 characters.

(3) 40% of traffic is narrative and 60% is data.

(4) Average message length is:

$$(.4)(1800) + (.6)(30,000) = 18,720 \text{ characters}$$

(5) Average multiplicity is 4:

for each message into the switch, 4 messages are generated by the switch.

18,720 (characters in) + 74,880 (characters out) = 93,600 characters throughput per message.

(6) Average message arrival per second = 1

(7) Sustained characters/second throughput = 93,600

(8) <u>Line Types</u>

Three line types are used whose characteristics are as follows:

a. Type 1: <u>Low Speed Line</u> (LSL) (300 bps)

Operates in AUTODIN modes II, IV and V. Uses ITA #2 code in JANAP-128 or ACP-127 formats.

b.   Type 2:   <u>M</u>edium <u>S</u>peed <u>L</u>ine (MSL) (2400 bps)

      Operates in AUTODIN modes II, IV and V. Uses
      ASCII code and JANAP-128 or ACP-127 formats.

c.   Type 3:   <u>H</u>igh <u>S</u>peed <u>L</u>ine (HSL) (9600 bps)

      Operates in AUTODIN modes I and III. Uses
      ASCII code and JANAP-128 formats.

Therefore: 2 codes, ITA #2 and ASCII, are used;
              2 formats, JANAP-128 and ACP-127 are used.

(9)   <u>Language Media Format (LMF)</u>

Three types of LMF's are assumed:

    (1)   card format

    (2)   paper tape format

    (3)   magnetic tape format

Any LMF may be used with any message.

Assume an even distribution of LMF type for all line
type messages.

(10)   <u>Channel Coordination</u>

Type 1 Line:

    <u>Data Receive</u> - recognize a steady mark (stop)
    signal as the idle pattern and a transition from
    mark to space as the start of the character frame.

    <u>Data Transmit</u> - two methods used;

    (1)   transmit data rhythmically with no external
         character stop

    (2)   transmit each character upon the receipt of
         a character step (release) pulse and maintain
         a steady mark condition upon completion of a
         character until the receipt of the next step
         pulse.

    Assumes that 1/2 of the Type 1 lines transmit
    with Method 2.

Type 2 Line:

Same as Type 1 line as far as coordination pro-
cedures are concerned.

Assume that 1/2 of the Type 2 lines transmit with
Method 2.

Type 3 Line:

<u>Data Receive</u> - data control characters used on a
84 character block basis. Receives idle pattern
for synchronization, verification, and channel
checking purposes.

<u>Data Transmit</u> - data control characters trans-
mitted with each 84 character block. During idle
periods, transmits the idle pattern.

(11) <u>Average Holding Time</u> ($\overline{HT}$) (by line speed)

$$HSL - \overline{HT} = 20 \text{ seconds/message}$$
$$MSL - \overline{HT} = 65 \text{ seconds/message}$$
$$LSL - \overline{HT} = 500 \text{ seconds/message}$$

(12) <u>Message Distribution</u> (by line speed)

Originated Line Speed	Messages Orig.	Terminated Line Speed			TOTALS
		HSL	MSL	LSL	
HSL	360	144	1224	72	1800
MSL	3060	1224	10404	612	15300
LSL	180	72	612	36	900
TOTALS	3600	1440	12240	720	18000

(13) <u>Distribution of Terminals</u> (by line speed)

Line Speed	Messages	Erlangs	Terminals	(GOS)
HSL	1440	8	18	.001
MSL	12240	221	245	.01
LSL	720	100	117	.01
TOTALS	18000	329	380	N/A

III-21

(14)   Message Accountability Distribution:

Minimum Acc.	20%
Intermediate Acc.	20%
Historical/Recover Acc. (H/R)	20%
Historical/Recovery/Traffic Acc. (H/R/T)	20%
Maximum Acc.	20%

(15)   Accountability Functions:

   a.   Minimum Accountability

Buffer storage for active processing

Maintains delivery queues

Maintains channel queues

Releases messages (removes from intransit storage) upon receipt of acknowledge by receiving equipment

Traffic statistics are kept, i.e.,

   message counts

   character counts

   statistics on acknowledgements (ACK's, NACK's)

   b.   Intermediate Accountability (Historical)

provides, in addition to minimum accountability facilities; historical records of predetermined processing events written to an on-line storage device. A journal of message events is kept.

   c.   Historical/Recovery Accountability (H/R)

in addition to the above, it provides recording of messages on an Intransit Storage file. Some measure of status ledgering is performed to maintain the storage structure of the messages (or blocks).

   d.   Historical/Recovery/Traffic Accountability (H/R/T)

in addition to the above, it provides recording of message blocks as received on separate recording medium. This allows retrieval and retransmission functions to be performed.

III-22

e. **Maximum Accountability**

all functions of above accountability procedures plus added redundancy and storage capabilities. Reference files are kept. Complete status ledgers redundantly stored.

(16) **Headers**

An average of three 84 character blocks per message required for headers.

(17) **Routing Indicators**

Data Messages - an average of 2 RI's per message.

Narrative Messages - an average of 7 RI's per message.

Average of 5 characters per RI (data messages)

Average of 20 characters per RI (narrative messages)

(18) **Routing Validation**

Data Messages - includes checking the originating station identifier and serial number, checking each RI against the RI Table and formatting the outgoing RI on all messages to other M/S's.

Narrative Messages - includes all of the processes for data messages but requires examining the textual content of each message to find the addresses.

(19) Retransmission of 5% of all 84 character blocks is assumed.

4.4 Circuit Switch Parameters

The Circuit Switch parameters are listed in Table III-3, below:

TABLE III-3

C/S SYSTEMS

PARAMETERS	SIZE			
# Lines	600	2400	4200	6000
Calls/Term/BH	5-8	4-7	3-6	2-5
Orig. Calls/BH	3000-4800	9600-16800	12600-25200	1200-30000
Orig. Erlangs	113E-180E	360E-630E	473E-945E	450E-1125E
Loop Orig.	45%	40%	35%	30%
Trunk Orig.	55%	60%	65%	70%
Register Hold Time	8-12 sec	8-12 sec	8-12 sec	8-12 sec

5.0 MODEL RUNS

This section contains a discussion of the way in which various design questions and options were investigated. In most cases, it was possible to evaluate the impact of a feature through the use of model switches and parameter values which were built into the model from the start. In some cases, changes to the model equations had to be made. In yet other cases, it was not possible or practical to investigate the impact of a feature since the need thereof at the time of model construction had not been predicted. There were also a few cases in which a modeling effort is not needed to determine the impact.

During the course of the study, questions concerning the effectiveness of several CPU features and options were derived. These design questions were:

   (1)   Data block move size

   (2)   Number of registers

   (3)   Level of registers

   (4)   Element size

   (5)   Base register

   (6)   Multiply instruction

   (7)   Memory-to-Memory transfer

   (8)   Literal to memory (greater than 1-bit)

   (9)   Shift/rotate

 (10)   Multiple level indirect addressing

 (11)   Search, link list, and compare encode instructions

Each of these features or instructions are discussed below.

   (1)   <u>Block Size for Block Oriented Instructions</u>

This question refers to the element addressing scheme proposed, in which an element is anything from a single bit to a block of many characters. Given such an addressing mode, investigate the impact of maximum block size on system performance.

There is no way to investigate this question in the context of the present model. The entire approach is radically different from addressing modes now available. Common subroutines and macros of the model were examined to determine if the impact could be localized to that point. It was determined that it could not – at best a small part of the impact would show up in existing subroutines and macros. For the most part,

the influence of this addressing mode is extremely diffuse.

(2) **Number of Registers**

The present model is based on 16 registers, half of which are used for indexing and addressing. Of these, it is felt that the indexing functions are the more serious ones. If the number of indexing registers were increased to above 8, say to 16, it is felt that there would not be much in the way of beneficial impact. Reduction of the index registers to, say, 4 would necessitate substitution of the indexing mode by explicit instructions about 30% of the time.

Therefore, this impact can be simulated by choosing the values for the indexing mode "surcharge", according to the following table:

Number of Index Registers	b	d	i	k
0	5	6	7	6
4	2.9	3.9	4.9	3.9
8	2	3	4	3

(3) **Sets of Registers**

This question relates to the number of hardware or software stored register sets. There is a switch for this purpose, VHSS, which was set to select hardware or software storage. In addition, the values of the program level change and return and the interrupt level change and return parameters were changed. While this impact is not absolutely modelable as phrased, it was felt that the following approximation would suffice.

Number of Sets of Hardware Registers	VHSS	JTR	DNIP
0	0		
1	1	25	20
2	1	20.2	15.2
3	1	16.8	11.84
4	1	14.4	9.44
5	1	12.8	7.8
6	1	11.7	6.7

Number of Sets of			
Hardware Registers	VHSS	JTR	DNIP
7	1	10.9	5.9
8	1	9	4

This can be accomplished in a simpler manner by substituting the following equations:

$$VHSS = ZNRS$$

$$JTR = 9 + (.7)^{(ZNRS-1)} * 16$$

$$DNIP = 4 + (.7)^{(ZNRS-1)} * 16$$

Where ZNRS = number of hardware register sets provided.

(4) <u>Element Size</u>

There is no direct way to implement this question. However, the context in which it occurs, is essentially covered by items 7 and 8 below.

(5) <u>Page Size</u>

The present model is based on a page size of 8,192 characters. There is a model switch, VPGA, which is intended to cover the use of an 8K page or no paging. For practical purposes, a page of 65,536 characters can be assumed to be an unpaged machine. This value (65,536) was also used as a standard size for a memory module. It was generally conceded that if page size were to be brought down to 512 characters, the system would not work. The impact of page size was expected to follow a loglinear behavior. There are three points which are readily available: 512 characters or less (infinite delay at minimal throughput), 8,192 characters (paging switch set to paging), 65,536 characters or greater (paging switch set to unpaged). The impact of paging was explored by setting the paging switch. A log-linear interpolation of the impact allowed us to determine the impact of page size.

(6) <u>Multiply Instruction</u>

The multiply instruction was available in the L3050 (the machine on which the model was based). It did not appear in any of the on-line paths modeled. Therefore, it can be safely assumed that the multiply instruction has no impact.

(7)   Memory to Memory Transfers

This is covered by use of the MOVE instruction switch, VMOV.

(8)   Literal to Memory Instruction Modes

This was modeled in two contexts: bit set and reset, and clear memory instruction. The bit set and reset instructions in one form or another are so essential that there can be no question of including them. The clear memory instruction was covered by the clear memory switch, VZRO.

(9)   Shift/Rotate Instruction

This was done by parameter setting as follows:

With shift/rotate   g = .05
Without shift/rotate   g = 3.00

(10)   Multiple Levels of Indirect Addressing

It was generally felt that the number of instances of multiple levels of indirect addressing would be, at most, 5% of the number of instances of single levels of indirect addressing. While the instances of multiple indirect addressing could not be isolated, the impact of indirect addressing versus no indirect addressing could be determined. 5% of this difference should account for the impact of multiple levels of indirect addressing. Therefore, the following parameter values are appropriate:

	c	d
With indirect addressing	3	3
Without indirect addressing	5	5

(11)   Search Instruction

There was no direct way to implement this question. It was determined that a search queue instruction is intimately tied up with the question of paging. If the machine is paged, this instruction would be of little use. If the machine is not paged, however, the impact is diffuse and hard to evaluate.

(12) Link List Instruction

There was a direct model switch for this instruction, VODB.

(13) Compare/Encode Instruction

There was no direct way of implementing this instruction. However, it should be noted that an instruction of this type was available in the L3050. Its absence in the model indicates the low frequency of usage. It was not expected to have any impact on the MP; however, it could have impact on the FE.

NOTE: This instruction was included in the design questions because it improves the efficiency of "path searching" in matrix control functions. It is of value only in systems in which the matrix map and search operations are performed entirely in software. Its value in these cases is unquestioned. The modeling effort was to determine what, if any, impact this instruction would have on the message switching operations.

(14) Memory Speed

Since memory speed buffer quantity trade-offs can result in lower total costs at higher memory speeds, this parameter was investigated as well for all models.

5.1 Summary of Parameters Which Remained to the Last Model Level

The following parameters had to remain to the last level of the model. Tables III-4 and III-5 show the runs and the parameter values to be used. Note that the indirect addressing question was covered by the index/indirect run (number of registers).

Run Objective	b*	c*	d*	g*	i*	k*	m*	ZNRS	VPGA	VMOV	VSRO	VODB
Best System	2	3	3	0.05	4	3	1	100	1	1	1	1
Worst System	8	5	11	3.00	10	10	1	1	0	0	0	0
Typical System	2	3	3	0.05	4	3	1	1	1	0	0	0
Number of Registers	8 4 0		8 4 0		8 4 0	8 4 0						
With Indirect	5,5.9,9	3	5,5.9,9	3.00	7,7.9,10	7,7.9,10	1	1	0	0	0	0
Without Indirect	5,5.9,8	5	8,8.8,11	3.00	7,7.9,10	7,7.9,10	1	1	0	0	0	0
Number of Register Sets	8	5	11	3.00	10	10	1	2-3	0	0	0	0
Paging/Not Paged	8	5	11	3.00	10	10	1	1	1	0	0	0
Move Instruction	8	5	11	3.00	10	10	1	1	0	1	0	0
Clear Instruction	8	5	11	3.00	10	10	1	1	0	0	1	0
Shift/Rotate	8	5	11	0.05	10	10	1	1	0	0	0	0
Link List	8	5	11	3.00	10	10	1	1	0	0	0	1
Memory Speed	8	5	11	3.00	10	10	0-3	1	0	0	0	0

TABLE III-4

Summary of Values for Message Processor Runs

* Indicates variables calculated as shown in the
applicable section of Paragraph 5.0.

Run Objective	b	c	d	g	i	m	ZNRS
Best System	3	3	3	0.05	4	1	100
Worst System	8	5	11	3.00	10	1	1
Typical System	2	3	3	0.05	4	1	1
Number of Registers	8 4 0		8 4 0		8 4 0		
With Indirect	5,5,9,8	3	5,5,9,9	3.00	7,7,9,10	1	1
Without Indirect	5,5,9,8	5	8,8,8,11	3.00	7,7,9,10	1	1
Number of Register Sets	8	5	11	3.00	10	1	2,3,4
Shift/Rotate	8	5	11	0.05	10	1	1
Indirect Addressing	5	5	11	3.00	10	0-2	1
Memory Speed	8	5	11	3.00	10	0-2	1

TABLE III-5

Summary of Values for Front-End Runs

6.0 MODEL REDUCTION PROCEDURE

6.1 Creation of Simplified General Model - CPM3

The first round of fixing was done using the common para-
meter values of Table III-6, resulting in model CPM2. As a re-
sult of this FIX pass, it was possible to compress the model to
the point where it would fit in core. In addition, all super-
fluous functions were eliminated.

6.2 Creation of Master Front-End Model - FEM2

The model selection switch, VMDL, is set to 0 which selects
the front-end model. In addition to this, every instance of the
function name "CLIN" is replaced with the name "CLIX" by use of
the substitute command. The bulk of the remaining modifications
consists of killing many functions which are superfluous to the
front-end model.

6.3 Creation of Master Message Processor Model - MPM2

The Master Model (CPM3) was used with the switch set to
select the message processing (MP) model. This resulted in model
MPM1. The front-end model residues were then killed by editing,
resulting in the master message processor model, MPM2.

6.4 Split Into Packet, Message, and Base Distribution Models

6.4.1 General

The second major step in the creation of the various models
was to set the values of additional parameters in the MP and FE
models to correspond to the packet switch MP and FE models (PMP
and PFE, respectively), the message switch MP and FE models (MMP
and MFE, respectively), and the base distribution MP and FE models
(BMP and BFE, respectively). A common parameter file was used
for the MP and FE models of each type to assure that the MP and
FE would be set to the same parameters. These files are given in
Table III-7. The packet switch file is called PAC1, the message
switch file is called MSG1, and the base distribution switch file
is called BAS1; the files are so identified by the column headings
in Table III-7.

In most cases, the parameter values chosen were design
options, hardware characteristics, or system characteristics.
A few of the values bear comment, however.

Source Model: CPS4,CPS5,CPS2,CPS1,CPS6,CPS3,CPS8,CPS7
Parameter File Name: CPM1

Resultant Model: CPM2
Resultant Numbers File Name: CPMV

	Parameter Name	Value	Meaning	Reference/Comment
1	A	2	Memory cycles for normal instruction	NEC Specification
2	CSCK	.25	Memory cycles stolen per character for MP-circuit switch transfer	All cycle stealing is 4 characters at a time within system
3	CSD	.25	Memory cycles stolen per character for ITS transfer	All cycle stealing is 4 characters at a time within system
4	CSJL	.25	Memory cycles stolen per character for journal transfer	All cycle stealing is 4 characters at a time within system
5	CSLD	.25	Memory cycles stolen per character for ledger device	All cycle stealing is 4 characters at a time within system
6	CSLF	1	Memory cycles stolen per character for line to FE	Front-end line unit buffers only one character at a time
7	CSMF	0	Memory cycles stolen per character for FE to MP	Shared memory operation between FE and MP
8	CSMM	.25	Memory cycles stolen per character for malfunction monitor	Ordinary internal channel
9	CSRF	.25	Memory cycles stolen per character for tape/disc transfers	Ordinary internal channel
10	CSRT	.25	Memory cycles stolen per character for retrieval device	Ordinary internal channel
11	CSSM	.25	Memory cycles stolen per character for circuit switch signaling channel	Ordinary internal channel
12	E	1	Memory cycles for long instructions	
13	F	2	Memory cycles for literal mode instructions	
14	GPL	7200	Holding time in seconds for permanent circuit switched lines	Lines are essentially always up (one hour)

TABLE III-6

Source Model: __CPS4,CPS5,CPS2,CPS1,CPS6,CPS3,CPS8,CPS7__   Resultant Model: __CPM2__

Parameter File Name: __CPM1__   Resultant Numbers File Name: __CPMV__

	Parameter Name	Value	Meaning	Reference/Comment
15	H	3	Memory cycles for special FE instruction	
16	HB1L	.000001	Seek factor for bubble memory	No bubble memory
17	HB1S	.000001	Seek factor for bubble memory	No bubble memory
18	HB2L	.000001	Seek factor for bubble memory	No bubble memory
19	HB2S	.000001	Seek factor for bubble memory	No bubble memory
20	HFA	50	Maximum seek time of reference disk	High speed disk
21	HFI	10	Minimum seek time of reference disk	High speed disk
22	HFR	25	Rotation time of reference disk	High speed disk
23	HGI	10	Minimum seek time for ITS disk	High speed disk
24	HGIW	10	Minimum seek time for ITS disc for ITS write	High speed disk
25	HGR	25	Rotation time for ITS device	
26	HGRW	25	Rotation time for ITS device write	
27	HJA	50	Maximum seek time for journal device	
28	HJI	10	Minimum seek time for journal device	
29	HJR	25	Rotation time for journal device	
30	HLA	50	Maximum seek time for ledger disk	
31	HLI	10	Minimum seek time for ledger disk	
32	HLR	25	Rotation time for ledger disk	

TABLE III-6 Continued

III-34

Source Model: CPS4,CPS5,CPS2,CPS1,SPS6,CPS3,CPS8,CPS7   Resultant Model: __CPM2__

Parameter File Name: __CPM1__   Resultant Numbers File Name: __CPMV__

	Parameter Name	Value	Meaning	Reference/Comment
33	HRA	50	Maximum seek time for retrieval disk	
34	HRI	10	Minimum seek time for retrieval disk	
35	HRR	25	Retrieval disk rotation time	
36	HTCL	0.0	Number of on-line tape controllers	No tapes
37	HTRL	0.0	Number of on-line tape transports	No tapes
38	ICCB	2048	Number of characters in ledger change buffer	Design option
39	ICDE	6	Number of characters per destination in destination table	Typical value
40	ICLK	10	Clock interrupt period in milliseconds	Design option
41	ICME	1024	Size of message exchange buffer	Design option
42	ICOB	12	Overhead characters per buffer block in MP	Design option
43	ICOF	8	Overhead characters per buffer block in FE	Design option
44	ICOQ	8	Overhead characters per queue block	Design option
45	ICPK	250	Packet length	AUTODIN-II value
46	ICQ	16	Total length of queue block	Design value
47	ICSD	512	Size of journal buffer	Design value
48	ICTB	100,000	Characters per tape block	No tapes
49	ICW	4	Number of characters per machine word	NEC
50	IGAP	.00001	Block gap time for tape	No tapes

TABLE III-6 Continued

Source Model: CPS4,CPS5,COS2,CPS1,CPS6,CPS3,CPS8,CPS7
Parameter File Name: CPM1

Resultant Model: CPM2
Resultant Numbers File Name: CPMV

	Parameter Name	Value	Meaning	Reference/Comment
51	ICSI	256	Number of characters per input segment	Design option
52	ILSI	3	Number of line blocks per input segment	Design option
53	ICP	84	Characters per line block for packet traffic	Typical value
54	ILSU	3	Line blocks per segment on output	Design option
55	IMON	100	Malfunction monitor period in milliseconds	Design option
56	INBL	1	Number of interrupts per crypto synch command	Typical value
57	INPK	1	Number of interrupts per output packet	Design option
58	INSG	3	Number of interrupts per output segment	Design option
59	IPKV	16	Number of characters overhead per packet on output	AUTODIN-II
60	IRQ	4	Number of RI's stored per queue block	Design option
61	ITRA	.000001	Characters in tape block	No tape
62	ITRB	.000001	Characters in tape block	No tape
63	ITRL	.000001	Characters in tape block	No tape
64	ITST	.000001	Tape start time	No tape
65	IWMC	2	Crypto MP-FE transfer field length	Typical value
66	IWRW	2	Crypto MP-FE transfer field length	Typical value
67	J	2	Memory cycle for short I/O operations	CPS design value
68	JBT	.004 (packet)	Number of bits tested in MP for message exchange	Typical value

TABLE III-6 Continued

Source Model: CPS4,CPS5,CPS2,CPS1,CPS6,CPS3,CPS8,CPS7   Resultant Model: CPM2

Parameter File Name: __CPM1__   Resultant Numbers File Name: __CPMV__

	Parameter Name	Value	Meaning	Reference/Comment
69	JCH	35	Effective number of characters in alphabet	Typical value
70	JCHP	8	Character pair pointers for retrieval	Typical value
71	JSC	.003 (packet)	State change tests per characters in message exchange	Typical value
72	JSL	3	Average length of tested sequence in message exchange	Typical value
73	JSQ	3	Number of sequences tested for simultaneously message exchange	Typical value
74	KHGI	.8	Seek time curve parameter	Typical value
75	LNAK	8	Circuit switch ACK message length	TRI-TAC value
76	LNCA	12	Circuit switch off hook message length	TRI-TAC value
77	LNCC	24	Circuit switch ringback message length	TRI-TAC value
78	LNCI	36	Circuit switch call initiate length	TRI-TAC value
79	LNCS	24	Circuit switch modem carrier signal message length	TRI-TAC value
80	LNOH	12	Circuit switch on-hook message length	TRI-TAC value
81	LNOR	24	Circuit switch on-hook/ringback message length	TRI-TAC value
82	LSED	2	Number of pseudo lines in system	Design option
83	LSVC	6	Number of service lines in system	Design option
84	LUTZ	.85	Average line utilization factor	Typical for peak traffic period

TABLE III-6 Continued

III-37

	Parameter Name	Value	Meaning	Reference/Comment
85	NCSB	1	Number of crypto synch commands received per block	Typical value
86	NMLA	0	Number of characters moved in RXLA	Design detail
87	NMLB	0	Number of characters moved in RXLB	Design detail
88	NMLD	0	Number of characters moved in RXLD	Design detail
89	NOP	0	Average number of openings/closings per second for circuit switched lines	All circuit switched lines are essentially permanent
90	NPR	2	Number of on-line printers	Typical value
91	NSWC	50	Number of switches in network	Expected sum of AUTODIN-I, AUTODIN-II, base switches
92	NXK	200	Internal parameter	
93	P1PC	0.0	Traffic parameter polling	Traffic type not in model
94	P5PC	0.0	Traffic parameter polling	Traffic type not in model
95	PCDI	.33	Probability of card input format	Typical value
96	PCDU	.33	Probability of card output format	Typical value
97	PSC	.2	Probability of a state change for a character in message exchange	Typical value
98	PSE	1.0	Probability that a line is secure	Typical value
99	T3	0	Polling message rate	No polling in model
100	TCPD	0	Expected number of characters to be padded at end of segment	Design option

TABLE III-6 Continued

Source Model: CPS4,CPS5,CPS2,CPS1,CPS6,CPS3,CPS8,CPS7
Parameter File Name: CPM1

Resultant Model: CPM2
Resultant Numbers File Name: CPMV

	Parameter Name	Value	Meaning	Reference/Comment
101	TCSP	3	Characters per RI separator group	Typical value
102	TCSQ	4	Characters per station sequence number	Typical value
103	TDSK	1,310,720	ITS transfer rate	CPS design
104	TLSC	12	Number of characters in a short control message	Typical value
105	TMF	1 billion	MP/FE transfer rate	Shared memory
106	TQH	.05	High pre-emptive message probability	Typical value
107	TRJL	1,310,720	Journal device transfer rate	CPS disc
108	TRLD	1,310,720	Ledger device transfer rate	CPS disc
109	VBIN	1	Disc parameters are a power of 2	CPS design
110	VBNK	1	I/O is not bank limited	CPS design
111	VCHA	0	System has character addressing	CPS design
112	VCIO	1	System has chained I/O	CPS design
113	VCOM	0	MP/FE not combined	CPS design
114	VCSI	0	There is a circuit switch interface	
115	VFE	0	Lines are encrypted	
116	VITS	1.5	Moving head disc for ITS	
117	VLED	1.5	Moving head disc for ledger	
118	VLUT	1	Line utilization is a parameter	

TABLE III-6 Continued

III-39

Source Model: CPS4,CPS5,CPS2,CPS1,CPS6,CPS3,CPS8,CPS7 Resultant Model: __CPM2__

Parameter File Name: __CPM1__ Resultant Numbers File Name: __CPMV__

	Parameter Name	Value	Meaning	Reference/Comment
119	VODX	1	Move is not required for message exchange	
120	VPKT	0	There is packet processing	
121	VPRI	1	RI's are packed to conserve space	
122	VROW	0	Lines are encrypted	
123	VSGT	0	Scatter/gather I/O	
124	VSIP	1	SIL/IDP are combined	
125	VSLI	0	Machine does not have a ledgering instruction	
126	VTDJ	1	Disc used for journal	
127	VTDR	1	Disc used for retrieval	
128	VTHP	0	Hardware parity check on lines	
129	VTLG	0	More than one character per word	
130	VTPS	1	CRC Hardware	

TABLE III-6 Continued

Source Models: MPM2,FEM2

Parameter Files: PAC1,MSG1,BAS1

Resultant Models: PMP1,BMP1,MMP1,PFE1,BFE1,MFE1

Resultant Numbers Files: PMPV,BMPR,MMPV,PFEV,BFEV,MFEV

	Name	FE/MP	PAC1 Packet Value	MSG1 Message Value	BAS1 Base Value	Meaning	Reference/Comment
1	CLA	FE	.05	.3	.3	MP cycle length as input to front-end	
2	CLB	FE	.02	.2	.2	MP cycle length as input to front-end	
3	CLC	FE	.001	.001	.001	MP cycle length as input to front-end	
4	CLFX	MP	.025	.05	.05	Length of FE cycle from point of view of MP	
5	HCKR	MP	0	41,943,040	41,943,040	ITS capacity in characters	No ITS on packet switch
6	HDCL	MP	0	2	2	Number of on-line controllers for ITS	No ITS on packet switch
7	HDCR	MP	0	4	6	Number of controllers for other storage functions	No discs on packet switch
8	HDJL	MP	0	1	1	Number of devices used for journaling	No discs on packet switch
9	HDRL	MP	0	6	8	Number of disc units on-line	No discs on packet switch
10	HFC	MP	0	128	128	Number of cylinders on reference disc	No discs on packet switch
11	HFS	MP	0	32	32	Number of logical sectors on reference disc	No discs on packet switch
12	HGC	MP	0	5	10	Number of cylinders on ITS for read operation	No discs on packet switch, ITS prepositioned
13	HGCW	MP	0	2	2	Number of cylinders on ITS for write operation	No discs on packet switch, ITS prepositioned
14	HGN	MP	0	2	2	Number of ITS units in system	No discs on packet switch
15	HGNR	MP	0	2	2	Effective number of ITS units for read operations	No discs on packet switch

TABLE III-7

Source Models: MPM2, FEM2

Parameter Files: PAC1, MSG1, BAS1

Resultant Models: PMP1, BMP1, MMP1, PFE1, BFE1, MFE1

Resultant Numbers Files: PMPV, BMPV, MMPV, PFEV, BFEV, MFEV

	Name	FE/MP	PAC1 Packet Value	MSG1 Message Value	BAS1 Base Value	Meaning	Reference/Comment
16	HGNW	MP	1	1	1	Effective number of ITS units for write operations	
17	HGS	MP	1	64	64	Number of logical sectors per ITS track	
18	HGSW	MP	1	128	128	Number of logical sectors per ITS track for writes	
19	HJC	MP	0	2	2	Effective number of cylinders on journal disc	
20	HJS	MP	0	128	128	Number of logical sectors on journal disc	
21	HJSA	MP	4	4	4	Number of segments to be written for various portions of ledgers	
22	HJSB	MP	3	3	3	Number of segments to be written for various portions of ledgers	
23	HL1	MP	2000	7000	7000	Cyclic ledger size parameters	
24	HL2	MP	20	150	200	Cyclic ledger size parameters	
25	HL3	MP	0	5000	5000	Cyclic ledger size parameters	
26	HLD	MP	1	2	2	Number of ledger discs	
27	HLJA	MP	1	2	2	Number of ledger copies written per cycle	
28	HLJB	MP	1	2	2	Number of ledger copies written per cycle	Ledgering is done to stand-by CPU in packet switch

TABLE III-7 Continued

Source Models: MPM2, FEM2
Parameters Files: PAC1, MSG1, BAS1

Resultant Models: PMP1, BMP1, MMP1, PFE1, BFE1, MFE1
Resultant Numbers Files: PMPV, BMPV, MMPV, PFEV, MFEV, BFEV, MFEV

	Name	FE/MP	PaC1 Packet Value	MSG1 Message Value	BAS1 Base Value	Meaning	Reference/Comment
29	HLJD	MP	1	2	2	Number of ledger copies written per cycle	
30	HRC	MP	1	128	128	Number of cylinders on retrieval disc	No retrieval in packet switch
31	HRS	MP	1	64	64	Number of sectors per track of retrieval disc	No retrieval in packet switch
32	HRTN	MP	0	4	6	Number of storage devices other than ITS	
33	ICB	FE/MP	256	64	64	Number of characters in MP buffer block	Design option
34	ICBF	FE	64	84	128	Number of characters in FE buffer block	Design option
35	ICFS	MP	1,000,000	1024	1024	Number of characters per sector (logical) of reference device	Design option, no reference in packet switch
36	ICHI	FE	4	4	4	Input characters accumulated in FE prior to interrupt	Design option
37	ICHS	MP	1,000,000	1024	1024	Number of characters per sector (logical) of retrieval device	Design option, no retrieval in packet switch
38	ICI	FE	4	4	4	Number of characters transferred per line interrupt in FE	
39	ICID	MP	4	4	4	Expected number of character pairs in block ID	Traffic parameter
40	ICJS	MP	256	256	256	Number of characters per logical sector of journal device	Design option, no journal in packet switch
41	ICK	MP	256	256	256	Number of characters per ITS chunk	Design option, no journal in packet switch

TABLE III-7 Continued

Source Models: MPM2, FEM2
Parameter Files: PAC1, MSG1, BAS1

Resultant Models: PMP1, BMP1, MMP1, PFE1, BFE1, MFE1
Resultant Numbers Files: PMPV, BMPV, MMPV, PFEV, BFEV, MFEV

	Name	FE/MP	PAC1 Packet Value	MSG1 Message Value	BAS1 Base Value	Meaning	Reference/Comment
42	ICKS	MP	256	256	256	Number of characters per logical disc sector of ITS	Design option, no journal in packet switch
43	ICL	FE/MP	64	84	128	Number of characters per line block	Typical value
44	ICRT	MP	0	32	32	Number of characters in a retrieval index entry	Typical value, no retrieval in packet switch
45	IJA	MP	0	140	140	SOH journal entry size	Typical values, no journal device on packet
46	IJB	MP	0	20	20	ETX journal entry size	
47	IJC	MP	0	20	20	HI-PRI journal entry size	
48	IMI	MP	50	1200	3000	Average number of messages on ITS during peak hour	
49	INDE	MP	500	5000	10,000	Number of destinations in destination tables	
50	IQB	MP	650	1800	4000	Total available queue blocks	Design parameters to be verified
51	ITSH	MP	10	300	5000	ITS holding time (average)	Typical value
52	JCOM	MP	0	2	5	Retrieval model complexity factor	No retrieval in packet switch
53	JXM4	MP	0	1	4	Number of type 4 retrieval requests per message retrieved	
54	NCNT	MP	4	8	2	Number of other switches connected to this switch	

TABLE III-7 Continued

Source Models: MPM2, FEM2
Parameter Files: PAC1, MSG1, BAS1

Resultant Models: PMP1,BMP1,MMP1,PFE1,BFE1,MFE1
Resultant Numbers Files: PMPV,BMPV,MMPV,PFEV,BFEV,MFEV

	Name	FE/MP	PAC1 Packet Value	MSG1 Message Value	BAS1 Base Value	Meaning	Reference/Comment
55	NCRT	MP	1	2	2	Number of circuit switch processors connected to this switch	
56	NRIT	MP	4000	6000	12,000	Number of RI's in routing table	
57	NTG	MP	4	8	2	Number of trunk groups in switch	
58	PAG1	MP	.01	180	180	Average age of message for retrieval type N, at time of retrieval	No retrieval in packet switch
59	PAG2	MP	.01	600	600	Average age of message for retrieval type N, at time of retrieval	No retrieval in packet switch
60	PAG3	MP	.01	600	3600	Average age of message for retrieval type N, at time of retrieval	No retrieval in packet switch
61	PAG4	FE/MP	.01	600	86,400	Average age of message for retrieval type N, at time of retrieval	No retrieval in packet switch
62	PCSI	MP	1	.5	.5	Probability that a message is received on a circuit switched line	
63	PCSU	MP	1	.5	.5	Probability that a message is transmitted on a circuit switched line	
64	PCON	FE	1	1	1	Probability of AUTODIN-I continuous mode	All medium speed lines
65	PER	MP	1	.1	.05	Percentage of circuit switched lines which are permanent	
66	POVB	MP	10^{-8}	10^{-6}	10^{-6}	Overflow probability for message buffers	
67	POVJ	MP	10^{-9}	10^{-7}	10^{-7}	Overflow probability for journal buffers	
68	POVQ	MP	10^{-9}	10^{-8}	10^{-8}	Overflow probability for queue buffers	

TABLE III-7 Continued

Source Models: MPM2, FEM2

Parameter Files: PAC1, MSG, BAS1

Resultant Models: PMP1, BMP1, MMP1, PFE1, BFE1, MFE1

Resultant Numbers Files: PMPV, BMPV, MMPV, PFEV, BFEV, MFEV

	Name	FE/MP	PAC1 Packet Value	MSG1 Message Value	BAS1 Base Value	Meaning	Reference/Comment
69	POVF	FE	.000001	.000001	.000001	Overflow probability for front-end buffers	
70	PTR1	FE/MP	.02	.05	.05	Percent of traffic to be retrieved by mode 1 retrieval	
71	PTR2	FE/MP	0	.01	.03	Percent of traffic to be retrieved by mode 2 retrieval	
72	PTR3	FE/MP	0	.005	.07	Percent of traffic to be retrieved by mode 3 retrieval	
73	PTR4	FE/MP	0	.001	.07	Percent of traffic to be retrieved by mode 4 retrieval	
74	TCEM	MP	8	27	35	Characters in EOM sequence	
75	TCER	MP	1	5	5	Characters in EOR sequence	
76	TCH	MP	150	150	250	Average header length	
77	TCR	MP	4.5	5	125	Average characters per RI	
78	TPEK	FE/MP	0	.4	.4	Peaking factor	
79	TRI	MP	1	2.7	3.5	Average number of RI's per message	
80	TZU	MP	.5	1	8	Number of lines printed or displayed per second	

TABLE III-7 Continued

III-46

(1) The packet switch had no discs, no retrieval and no explicit intransit store other than core. The values chosen for the disc unit parameters of the packet switch were such as to eliminate that function from the model.

(2) The packet switch was conceived of as having a cyclic ledger, which was "written" to the standby CPU every cycle.

(3) Buffer block sizes were chosen to be optimum for all three types of systems, taking into account necessary compromises between efficiency, line speed, delay, etc.

(4) An explicit journal entry function did not exist in the packet switches.

(5) There was no retrieval in the packet switch, however, parameters PAG1-PAG4 had to be set to finite values to avoid algebraic expressions of the form 0/0.

(6) Design overflow probabilities for the packet switch were set very low because these switches do not have an intransit store or an overflow store.

(7) Mode 1 retrievals in the packet switch were intended to simulate the effect of line block retransmission processing.

6.4.2 Changes in Traffic Parameters

The actual construction of the six models was done in two phases. The first phase was used to fix the models to the values of Table III-7. The second phase was used to fix traffic parameters, as given in Table III-8 (files PAC2, MSG2, BAS2). During the model installation, it was decided that many traffic parameters could be eliminated since they were not truly independent of other parameters. Therefore, derivable parameters were replaced by equations. This file, TRAF, was merged with the CPSN Master Models.

6.4.3 Creation of Packet Switch MP Model

The packet switch MP model (PAC1) was developed from the MP model (MPM2) by setting the values of the parameters given in Table III-7, column 1. The resulting model was called PMP1. The derived value file was called PMPV. Spurious functions were deleted or edited as appropriate. The resulting model was called PMP2. The revised traffic parameters/equations were merged to PMP2, to create model PMP3. PMP3 was then fixed to

PARAMETER NAME	PAC2	BAS2	MSG2	MEANING
LIN1	30.00 (7.5)	210.00 (52.5)	70.00	Number of Mode I Lines (Front-End)
LIN2	0.00	160.00 (40)	45.00	Number of Mode II Lines (Front-End)
LIN4	0.00	0.00	0.00	Number of Mode IV Lines
LIN5	0.00	20.00 (5.0)	27.00	Number of Mode V Lines (Front-End)
LINH	10.00 (2.5)	5.00 (1.25)	4.00	Number of Packet Trunks (Front-End)
LINO	10.00 (2.5)	5.00 (1.25)	4.00	Number of Low Level Packet Trunks (Front-End)
LTS1	150.00	150.00	150.00	Average Line Speed of Mode I Lines
LTS2	30.00	30.00	30.00	Average Line Speed of Mode II Lines
LTS4	20.00	20.00	20.00	Average Line Speed of Mode IV Lines
LTS5	75.00	75.00	75.00	Average Line Speed of Mode V Lines
LTSH	2375.00	2375.00	2375.00	Average High Level Packet Lines
LTSO	2375.00	1200.00	1200.00	Average Low Level Packet Lines
NCST	20.00	10.00	8.00	Percent of Circuit Switched Trunks
PRBD	.01	.05	.07	Percent of Baudot Input Messages
PRSC	.04	.85	.90	Percent of ASCII Input Messages
PRFR	0.00	0.00	0.00	Percent of Free Form Traffic
PRLG	.75	.50	.25	Percent of Long Messages

TABLE III-8

PARAMETER NAME	PAC2	BAS2	MSG2	MEANING
PRSR	.25	.50	.75	Percent of Short Messages
PRPK	.95	.1	.03	Percent of Messages Input in Packet Format
T1	0.00	0.00	0.00	Single Packet Inquiry Response Traffic
TDMI	1.00	2.4	1.60	Average Deliveries per Message
TLSF	1.00	1.00	1.00	Average Length of FE Message
TLSL	30000.00	30000.00	30000.00	Average Length of Long Message
TLSS	1800.00	1800.00	1800.00	Average Length of Short Message
TM4	1.00	2.40	1.60	Short Baudot Multiplicity
TM5	1.00	1.00	1.00	Long Baudot Multiplicity
TM6	1.00	1.00	1.00	Baudot FF Multiplicity
TM7	1.00	2.40	1.60	ASCII Short Multiplicity
TM8	1.00	1.00	1.00	ASCII Long Multiplicity
TM9	1.00	1.00	1.00	ASCII FF Multiplicity
TRM3	1.00	2.00	1.00	Number of Messages that Satisfy Type 3 Retrieval Criteria
TRM4	1.00	3.50	1.20	Number of Messages that Satisfy Type 4 Retrieval Criteria
TP44	0.00	.05	.05	Baudot Short to Baudot Short Non-Conversion Ratio

TABLE III-8 (Continued)

PARAMETER NAME	PAC2	BAS2	MSG2	MEANING
TP47	1.00	.80	.80	Baudot Short to ASCII Short Conversion Ratio
TP4S	0.00	.15	.15	Baudot Short to Packet Short Conversion Ratio
TP55	0.00	.05	.05	Baudot Long to Baudot Long Non-Conversion Ratio
TP58	1.00	.80	.80	Baudot Long to ASCII Long Conversion Ratio
TP5L	0.00	.15	.15	Baudot Long to Packet Long Conversion Ratio
TP66	0.00	1.00	1.00	Baudot Free Form to Baudot Free Form Non-Conversion Ratio
TP69	1.00	0.00	0.00	Baudot Free Form to ASCII Free Form Conversion Ratio
TP6F	0.00	0.00	0.00	Baudot Free Form to Packet Free Form Conversion Ratio
TP74	0.01	.05	.05	ASCII Short to Baudot Short Conversion Ratio
TP77	0.99	.80	.80	ASCII Short to ASCII Short Non-Conversion Ratio
TP7S	0.00	.15	.15	ASCII Short to Packet Short Conversion Ratio
TP85	0.00	0.00	0.00	ASCII Long to Baudot Long Conversion Ratio
TP88	1.00	.15	.15	ASCII Long to ASCII Long Non-Conversion Ratio
TP8L	0.00	.85	.85	ASCII Long to Packet Long Conversion Ratio
TP96	0.00	.05	.05	ASCII Free Form to Baudot Free Form Conversion Ratio

TABLE III-8 (Continued)

PARAMETER NAME	PAC2	BAS2	MSG2	MEANING
TP99	1.00	.80	.80	ASCII Free Form to ASCII Free Form Non-Conversion Ratio
TP9F	0.00	.15	.15	ASCII Free Form to Packet Free Form Conversion Ratio
TPF6	0.00	0.00	0.00	Packet Free Form to Baudot Free Form Conversion Ratio
TPF9	0.0	.5	.5	Packet Free Form to ASCII Free Form Conversion Ratio
TPFF	1.0	.5	.5	Packet Free Form to Packet Free Form Non-Conversion Ratio
TPL5	0.0	0.0	0.0	Packet Long to Baudot Long Conversion Ratio
TPL8	0.0	.75	.9	Packet Long to ASCII Long Conversion Ratio
TPLL	1.0	.25	.1	Packet Long to Packet Long Non-Conversion Ratio
TPS4	0.0	.05	.05	Packet Short to Baudot Short Conversion Ratio
TPS7	0.0	.85	.9	Packet Short to ASCII Short Conversion Ratio
TPSS	1.0	.10	.05	Packet Short to Packet Short Conversion Ratio
IBTI	8	48	48.00	Number of Change Buffer Characters per Input Message (Non-Cyclic Ledger)
IBTU	4	12	12.00	Number of Change Buffer Characters per Output Message (Non-Cyclic Ledger)
LNTS	16	16	16.00	Length of a Test Signaling Message
LSED	8	40	20.00	Number of Pseudo Lines in the System
MCMS	6000 (800)	40,000 (2,000)	24000.00 (3000)	Total Number of Characters in Miscellaneous Tables (Front-End)

TABLE III-8 (Continued)

III-51

PARAMETER NAME	PAC2	BAS2	MSG2	MEANING
MOVR	0.0	0.0	0.00	Total Number of Characters in Overlay Buffer
MPRO	35,000 (8,000)	170,000 (12,000)	140,000 (10,000)	Size of Program Space in Characters (Front-End)
NCDS	20	30	30	Number of Characters per Destination in Destination Tables
NCRI	5.4	5.4	5.4	Number of Characters per RI in RI's Tables
NLTB	30 (10)	140 (12)	140.0 (12)	Number of Characters per Line in Line Tables (Front-End)
NPSU	30	140	140.0	Number of Characters per Pseudo Line in Line Tables

TABLE III-8 (Continued)

the values of Table III-8, spurious functions eliminated, resulting in a new model, called PMP4.

6.4.4 Creation of Message Switch MP Model

The process was analogous to that used to create the packet switch MP model. MSG1.PAR was applied to MPM2.FUN to create MMP1.FUN and the derived value file MMPV.PAR. MMP1.FUN was edited to create MMP2.FUN. MMP2.FUN was fixed using parameter file MSG2.PAR resulting in model MMP3.FUN and derived value file MMPW.PAR. MMP3.FUN was then edited to create MMP4.FUN.

6.4.5 Creation of Base Distribution Switch MP Model

This is shown schematically below.

MMP2.FUN   BAS1.PAR   ⟶   BMP1.FUN, BMPV.PAR

BMP1.FUN   Edit   ⟶   BMP2.FUN

BMP2.FUN   BAS2.PAR   ⟶   BMP3.FUN, BMPW.PAR

BMP3.FUN   Edit   ⟶   BMP4.FUN

6.4.6 Creation of Packet Switch Front-End Model

FEM2.FUN   PAC1.PAR   ⟶   PFE1.FUN, PFEV.PAR

PFE1.FUN   Edit   ⟶   PFE2.FUN

PFE2.FUN   PAC2.PAR   ⟶   PFE3.FUN, PFEW.PAR

PFE3.FUN   Edit   ⟶   PFE4.FUN

6.4.7 Creation of Message Switch Front-End Model

FEM2.FUN   MSG1.PAR   ⟶   MFE1.FUN, MFEV.PAR

MFE1.FUN   Edit   ⟶   MFE2.FUN

MFE2.FUN   MSG2.PAR   ⟶   MFE3.FUN, MFEW.PAR

MFE3.FUN   Edit   ⟶   MFE4.FUN

6.4.8 Creation of Base Distribution Front-End Model

FEM2.FUN   BAS1.PAR   ⟶   BFE1.FUN, BFEV.PAR

BFE1.FUN   Edit   ⟶   BFE2.FUN

BFE2.FUN   BAS2.PAR   ⟶   BFE3.FUN, BFEW.PAR

7.0   PERFORMANCE RUN RESULTS - MESSAGE PROCESSORS

7.1   Definition of Capacity

If substantive comparisons of performance are to be made for various options tested, a quantitative measure of performance must be devised. A measure, called "capacity" was chosen which is defined as:

$$C = \frac{R_m}{S \, dR}{}_{0}$$

Where: C   is the system's capacity in $(transactions/sec)^2$

        S   is the reciprocal of transaction processing delay, (called the <u>Subjective Rate</u>)

        R   is the rate at which the transactions are presented to the system (<u>Objective Rate</u>)

        R_m is the transaction arrival rate at which the delay is infinite.

The relation between throughput and delay is called the throughput-delay curve. The above curve is called the "objective-subjective" rate curve. The justification for this measure of capacity is to be found in a copyrighted document "Of Systems and Their Idealization" by Boris Beizer. Briefly, the subjective rate is the rate that the user of the system experiences. The objective rate is the rate at which the system processes transactions. It can be shown that under general assumptions, the sum of the objective and subjective rates are almost constant and equal to the reciprocal of the CPU instructions executed per transactions. The above measure of capacity is justified heuristically on the following grounds.

(1) Doubling the subjective rate doubles the system's capacity.

(2) Doubling the objective rate doubles the system's capacity also.

(3) Grosch's square law for performance versus memory speed for ideal systems can be derived from it.

A formal derivation of these concepts can be found in the aforementioned publication, available from the author.

7.2   Summary of Results for Message Switch MP

7.2.1   General

The summary results for the message switching system message processor are shown in Tables III-9 and III-10.   Initial de-bugging runs were done on the "typical system".   The typical system is comparable to the higher quality CPU's available today for message switching.   Initial runs were made to determine the impact of the number of front-ends on the behavior of the message processor.   That impact was almost undetectable.   Therefore, all MP runs were made with 4 front-ends, as a typical value.

The chart shows the case, the run identifications, the capacity, the improvement in capacity as compared to the worst system, the point at which the system "blows up" (actually, cycle length = 50-1000 seconds), and the zero traffic behavior (actually, .00001 msg/sec).   Once the typical system had been analyzed, the "best" and "worst" systems were run.

7.2.2   "Best, Typical, Worst"

There was hardly any difference between the "best" and the "typical" system from the point of view of overall capacity. The "best" system was 149% better than the worst, while the typical was 142% better than the worst.

7.2.3   Impact of Indexing and Indirect Addressing

Indirect addressing and the number of index registers were the single most significant feature investigated.   Taken separately, indirect addressing accounted for a 10% improvement. Going from 0 to 8 index registers, without indirect provided a 27% improvement.   However, going to 8 index registers with indirect addressing resulted in 38% improvement over the worst system.   This by far accounted for the bulk of the performance improvement of the best over the worst system.   Introducing two levels of indirect addressing, based on the analysis of the previous section would result in an insignificant further improvement.   On the other hand, increasing the number of registers to 16 would result in a further 10-20% improvement.   Based on the result for the MP, then, it is strongly recommended having a minimum of 16 hardware index (actually generalized) registers with indirect addressing.

7.2.4   Paging

Paging was seen to have a major impact on the message MP performance, second only to indexing and indirect.   The indications are that the message switch MP should not be paged, or should have a large page size, e.g., 65,536 characters.

SUMMARY OF RUN RESULTS FOR MESSAGE SWITCH, MESSAGE PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC) 2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
Typical System		1101-1106	3.4609	142.00%	.25934	3.30
Best System		1201-1206	3.5691	149.60%	.25693	3.38
Worst System		1301-1306	1.4299	0.00%	.34515	1.90
# of Register Sets						
1 (Worst)		1301-1306	1.4299	0.00%	.34575	1.90
2		1250-1251	1.4390	.71%	.34297	1.91
3		1260-1261	1.4470	1.20%	.34163	1.91
Paging		1401-1406	1.6482	15.30%	.32441	2.05
Move Instruction		1501-1502	1.4375	.53%	.34515	1.92
Clear Memory		1600-1604	1.4408	.77%	.34471	1.90
Link List		1650-1651	1.4308	.07%	.34491	1.91
Shift		1700-1704	1.6498	15.40%	.32004	1.99
Index/Indirect						
8 Registers No Indirect		1780-1782	1.8116	26.70%	.30914	2.34
4 Registers No Indirect		1785-1787	1.6478	15.24%	.31888	2.21

TABLE III-9

SUMMARY OF RUN RESULTS FOR MESSAGE SWITCH, MESSAGE PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)^2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
0 Registers No Indirect		1301-1306	1.4299	0.00%	.34515	1.90
8 Registers With Indirect		1750-1752	1.9546	36.70%	.30651	2.40
4 Registers With Indirect		1760-1762	1.7876	25.02%	.31611	2.40
0 Registers With Indirect		1775-1777	1.9586	2.01%	.34129	1.93

TABLE III-9 (Continued)

III-57

SUMMARY OF RUN RESULTS FOR MESSAGE SWITCH, MESSAGE PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)^2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
Memory Speed						
.000001		1940-1942	33.419	2237.00%	.22538	64.00
.25		1950-1952	6.545	357.00%	.25074	6.80
.35355		1800-1801	4.766	233.00%	.26154	5.60
.40825		1820-1821	4.126	188.00%	.26724	4.80
.447214		1830-1831	3.748	162.00%	.27131	4.00
.500		1840-1842	3.348	134.17%	.27682	3.85
.57735		1850-1852	2.862	100.19%	.28489	3.26
.70711		1860-1862	2.254	57.62%	.30018	2.80
1.000		1301-1306	1.430	0.00%	.34515	1.90
1.414214		1870-1872	.793	- 44.50%	.41796	1.33
1.732054		1880-1882	.547	- 61.77%	.47420	1.11
2.000		1890-1892	.433	- 69.75%	.52174	1.00
2.236068		1900-1902	.360	- 74.80%	.56372	.86
2.44945		1910-1912	.309	- 78.40%	.60175	.80
2.82843		1920-1922	.242	- 83.04%	.66947	.70
3.000		1930-1932	.219	- 84.67%	.70021	.62

TABLE III-10

7.2.5 Shift Instruction

The 15% impact of this instruction makes proper shifts mandatory.

7.2.6 All the Rest

The aggregate effect of the rest of the features investigated was under 2%. It would seem then, that from the point of view of message switch MP performance, the MOVE, CLEAR MEMORY, and LINK-LIST, instructions are of marginal utility. Similarly, providing a duplicate set of hardware registers to accommodate interrupt and program level shifts is of marginal utility in the MP.

7.2.7 Impact of Memory Speed on Message Switch MP Performance

Table III-10 shows the effect of memory speed. This was done by varying the memory speed for the "worst system" over a range which encompassed the unrealistic (.001 nanoseconds) to the abominable. The values chosen are intended to allow easy tracking of square-law behavior. It can be seen that this is the case over the region from 1 to 3 microseconds. At higher memory speeds, the system becomes increasingly transfer bound in the low traffic regions, thus removing some of the square law performance improvement effects. Since cost does not generally double as memory cycle time is halved, there is a strong indication that it would be worthwhile to target higher memory speeds, in the 300 to 700 nanosecond region. Note that in making this evaluation, the impact of memory speed on total memory required should not be ignored: this can more than buy back the extra cost of higher speed memories.

It is interesting to note that for all but the very fastest (.001 nanosecond) memory, that the system is ultimately CPU bound rather than transfer bound. This can be seen from an examination of the detail reports (see particularly the base switch MP data for this). Thus, for the message switch MP, it would seem that the present day high quality peripheral memories (such as were assumed in the model) are adequate and that no special development in this area is warranted. That is, exotica such as bubble memories will have to hack it on a price basis rather than on a price-performance basis, as the additional performance will probably not be needed. The programmable speed feature of a bubble memory would still allow a design which was free of processing gaps over most traffic conditions. This could result in improvements for small switches.

7.3 Summary of Results for Base Switch MP

Table III-11 shows the impact of the various options on the base distribution switch MP. Since the base switch is more CPU

SUMMARY OF RUN RESULTS FOR BASE DISTRIBUTION SWITCH, MESSAGE PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)²	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
Typical System		2100-2105	1.26852	164.8%	.4024	2.07
Best System		2200-2206	1.29459	170.3%	.3993	2.17
Worst System		2300-2306	.47894	0.00%	.5637	1.22
# of Register Sets						
1 (Worst)		2300-2306	.47894	0.00%	.5637	1.22
2		2250-2253	.48129	.49%	.5617	1.22
3		2260-2263	.48284	.815%	.5603	1.22
Paging		2400-2403	.56737	18.5%	.5431	1.33
Move Instruction		2500-2503	.48009	.24%	.5639	1.22
Clear Memory		2600-2603	.48132	.496%	.5636	1.22
Link List		2650-2653	.47921	.05%	.5637	1.22
Shift		2700-2703	.58679	22.50%	.5150	1.22
Index/Indirect						
8 Registers No Indirect		2780-2783	.68626	43.3%	.5018	1.48
4 Registers No Indirect		2785-2788	.62410	30.3%	.5200	1.40
0 Registers No Indirect		2300-2306	.47894	0.00%	.5637	1.22

TABLE III-11

SUMMARY OF RUN RESULTS FOR BASE DISTRIBUTION SWITCH, MESSAGE PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)²	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
8 Registers With Indirect		2750-2753	.69863	45.87%	.4981	1.50
4 Registers With Indirect		2760-2763	.63458	32.50%	.5162	1.41
0 Registers With Indirect		2775-2778	.48654	1.59%	.5596	1.22

TABLE III-11 (Continued)

bound than the message switch, an intensification of the results was expected. Note that the capacities are not directly comparable with the message switch capacities since the base switch is working with a larger average message size. The following observations are appropriate:

(1) The impact of additional registers sets, the MOVE, CLEAR MEMORY, LINK LIST instructions was still insignificant.

(2) Paging, shifts, and index registers were relatively more important.

(3) Indirect addressing was not as significant.

The memory speed situation is shown in Table III-12. Fewer runs were made since the same general picture emerged. The improvement for higher speed memory was greater, as expected.

It was surprising to find that the impact of retrieval, which had been expected to create significant transfer binding in the base switch, did not materialize. It could be argued that the quantity of retrievals was under estimated. This could be the case for some base switches, but it is not likely for the typical base switch. Again, the peripherals are not the problem, and the effort must be concentrated on the CPU.

7.4 Summary of Results for Packet Switch Message Processor

The summary results for the packet switch MP are shown on Tables III-13 and III-14. Again the picture is similar to that obtained for the message and base distribution switches.

The number of register sets was more significant for the packet switch, to the point where multiple register sets could be considered. Paging was more significant (48%), as were shifts (33%), and indexing and indirect. The clear memory instruction is on the border of significance at 8% improvement over the "worst" system. The overall results are not unexpected because the packet switch is mostly CPU bound - this would lead to a relatively greater emphasis on CPU facilities than would be observed in the partially transfer bound message and base switch MP's.

Memory speed impact was generally more rewarding as expected in a CPU bound system. It was not possible to obtain solutions for memory speeds of .001 nanoseconds because of GASM host computer exponent overflow and underflow problems. Accordingly, the memory speed was cut to .01 microseconds, which is still effectively infinitely fast by today's standards.

SUMMARY OF RUN RESULTS FOR BASE DISTRIBUTION SWITCH, MESSAGE PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC) ^{2}	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
Memory Speed						
.000001	Infinitely Fast Memory	2940-2943	10.634	2120.40%	.3216	29.44
.25	Best Likely Memory	2950-2953	2.552	533.00%	.3576	4.34
.500	Most Likely Memory	2840-2843	1.356	183.00%	.4146	2.28
1.000	Standard Memory	2300-2306	.479	0.00%	.5637	1.22
2.000	Worst Feasible Memory	2890-2893	.163	- 66.02%	.8645	.720

TABLE III-12

SUMMARY OF RUN RESULTS FOR PACKET SWITCH, MESSAGE PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)^2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
Typical System		3100-3106	31.1080	528.19%	.0479	8.9380
Best System		3200-3206	34.3252	593.16%	.0445	9.2000
Worst System		3300-3306	4.9520	0.00%	.1286	3.1000
# of Register Sets						
1 (Worst)		3300-3306	4.9520	0.00%	.1286	3.1000
2		3250-3254	5.0410	9.78%	.1265	3.1580
3		3260-3264	5.1040	11.15%	.1251	3.1780
Paging		3400-3404	7.3120	47.66%	.1085	4.4070
Move Instruction		3500-3504	4.9520	0.00%	.1286	3.1449
Clear Memory		3600-3604	4.9560	7.93%	.1286	3.1389
Link List		3650-3654	4.9520	0.00%	.1286	3.1388
Shift		3700-3704	6.6030	33.34%	.1040	3.4769
Index/Indirect						
8 Registers No Indirect		3780-3783	7.6230	66.01%	.0990	3.8992
4 Registers No Indirect		3785-3789	6.6490	34.30%	.1075	3.6352
0 Registers No Indirect		3300-3306	4.9520	0.00%	.1286	3.1000

TABLE III-13

SUMMARY OF RUN RESULTS FOR PACKET SWITCH, MESSAGE PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC) 2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
8 Registers With Indirect		3750-3754	8.0550	75.41%	.0961	4.0636
4 Registers With Indirect		3760-3764	7.0010	41.38%	.1044	3.7783
0 Registers With Indirect		3775-3780	5.1809	12.82%	.1252	3.2448

TABLE III-13 (Continued)

SUMMARY OF RUN RESULTS FOR PACKET SWITCH, MESSAGE PROCESSOR

CASE Memory Speed	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)^2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
.01	Infinitely Fast Memory	3940-3944	10407.7500	210,000.00%	.0083	318.2000
.25	Best Likely Memory	3950-3954	72.1417	356.82%	.0349	12.6900
.50	Most Likely Memory	3840-3844	19.3963	291.69%	.0644	6.3220
1.00	Standard Memory	3300-3306	4.9520	0.00%	.1286	3.1000
2.00	Worst Possible Memory	3890-3894	1.2398	- 74.96%	.2585	1.5530

TABLE III-14

8.0 PERFORMANCE RUN RESULTS - FRONT-END PROCESSORS

8.1 General

The front-end processor models were obtained from the Master Model as described above. However, a substantial amount of modification was required to achieve this, making the FEM1 model the basic front-end model. The capacity, as before, was obtained as the integral of the subjective rate. However, the subjective rate here is not simply the reciprocal of the delay shown in the reports. In order to make meaningful capacity comparisons to the capacity of the associated MP's, a transformation is required. One-half of the reciprocal of the front-end minor cycle length is called the "message delay". This is true in that it is the delay that will be seen by the last character of the message (EOM). However, this leads to subjective rates in the thousands. This rate can be interpreted as the rate at which groups of four characters are serviced. Dividing this by the average message length, a subjective rate in messages per second is obtained which is more compatible to that used by the MP.

All runs were performed with the number of front-ends set to 1 (LTCN = 1), so that the capacity of one front-end could be measured. Approximately 4 are required in most cases.

The following parameters did not show up in the final front-end models: paging switch, data move switch, clear memory switch, and link list instruction switch. Consequently, runs related to determining the impact of these features were not done. The non-appearance of the k instruction was a question of the paths selected by the model input parameters.

The other parameters (switches), apparently were not incorporated in the front-end model because the opportunity to do so did not reveal itself. That is, nowhere on the paths examined did it appear possible to incorporate such switches. It can, therefore, be said that it is unlikely that the impact of these features would be significant.

8.2 Performance Results for the Message Switch Front-End

The results for the message switch front-end are shown in Table III-15. The results are similar to that which was obtained for the MP, except that duplicate register sets and indirect addressing had a somewhat better payoff.

8.3 Performance Results for the Base Distribution Switch Front-End

These results are shown in Table III-16. The results are almost identical to that obtained for the message switch front-end.

SUMMARY OF RUN RESULTS FOR MESSAGE SWITCH, FRONT-END PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC) 2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
Typical System		4100-4103	1.5286	215.80%	.0436	.9510
Best System		4200-4203	2.1491	344.00%	.0367	1.1504
Worst System		4300-4303	.4840	0.00%	.0818	.5539
# of Register Sets						
1 (Worst)		4300-4303	.4840	0.00%	.0818	.5539
2		4250-4253	.5862	21.10%	.0743	.6173
3		2260-2263	.5865	21.10%	.0743	.6176
Shift		4700-4703	.5528	14.20%	.0729	.5765
Index/Indirect						
8 Registers No Indirect		4780-4783	.6789	40.26%	.0686	.6603
4 Registers No Indirect		4785-4788	.6090	25.82%	.0726	.6244
0 Registers No Indirect		4300-4303	.4840	0.00%	.0818	.5539
8 Registers With Indirect		4750-4753	.7751	60.14%	.0642	.7091
4 Registers With Indirect		4760-4763	.6954	43.67%	.0681	.6725

TABLE III-15

SUMMARY OF RUN RESULTS FOR MESSAGE SWITCH, FRONT-END PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)²	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
0 Registers With Indirect		4775-4778	.5425	12.07%	.0773	.5903
Memory Speed						
.010	Infinitely Fast Memory	4940-4943	384,284.9140		.0000823	554.6300
.250	Best Likely Memory	4950-4953	7.4055	1429.90%	.0204	2.2180
.500	Most Likely Memory	4840-4843	1.8519	282.60%	.0409	1.1090
1.000	Standard Memory	4300-4303	.4840	0.00%	.0818	.5539
2.000	Worst Possible Memory	4890-4893	.1164	- 75.95%	.1635	.2766

TABLE III-15 (Continued)

SUMMARY OF RUN RESULTS FOR BASE DISTRIBUTION SWITCH, FRONT-END PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)^2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
	Typical System	5100-5103	.5369	228.38%	.0291	.6072
	Best System	5200-5203	.7668	369.00%	.0246	.7179
	Worst System	5300-5303	.1635	0.00%	.0548	.3466
	# of Register Sets					
	1 (Worst)	5300-5303	.1635	0.00%	.0548	.3466
	2	5250-5253	.1970	20.50%	.0498	.3828
	3	5260-5263	.1971	20.60%	.0498	.3829
	Shift	5700-5703	.1870	14.40%	.0498	.3608
	Index/Indirect					
	8 Registers No Indirect	5780-5783	.2315	41.59%	.0460	.4147
	4 Registers No Indirect	5785-5788	.2071	26.67%	.0487	.3917
	0 Registers No Indirect	5300-5303	.1635	0.00%	.0548	.3466
	8 Registers With Indirect	5750-5753	.2669	63.24%	.0430	.4488
	4 Registers With Indirect	5760-5763	.2379	45.57%	.0457	.4219

TABLE III-16

SUMMARY OF RUN RESULTS FOR BASE DISTRIBUTION SWITCH, FRONT-END PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY 2 (MSG/SEC)	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
0 Registers With Indirect		5775-5778	.1835	12.23%	.0518	.3692
Memory Speed						
.01	Infinitely Fast Memory	5940-5943	1452.8000		.00055	34.6800
.25	Best Likely Memory	5950-5953	2.6521	1522.00%	.0137	1.3870
.50	Most Likely Memory	5840-5843	.6634	305.70%	.02741	.6934
1.00	Standard Memory	5300-5303	.1635	0.00%	.0548	.3466
2.00	Worst Possible Memory	5890-5893	.0409	- 75.00%	.1097	.1732

TABLE III-16 (Continued)

8.4 Performance Results for the Packet Switch Front-End

These results are shown in Table III-17. The results are almost identical to that obtained from the message switch front-end.

9.0 SUMMARY OF PERFORMANCE RESULTS

Table III-18 shows the development of the relative weights used to evaluate the impact of each feature investigated. The run results for the "typical system" were used to obtain typical configurations. This was done by selecting the input rates at which a delay of 2, 5, or .2 seconds was obtained for each of the message switches, base switches, and packet switches respectively. Given this traffic level, the required number of front-ends were obtained. From this, the typical memory requirements were developed. Most important, however, was the ratio of MP's to front-ends.

These data were incorporated into Table III-19 and combined with the expected deployment percentages (by sites) for each of the three switch models (system weight). This was combined with the weights of Table III-18 and renormalized to obtain the normalized weights for each of the 6 models. The percentile improvement accruing from each feature was multiplied by the normalized weights and added feature by feature to give the average impact in the last column.

9.1 Other Facts Noticed

9.1.1 Character Manipulation

The nature of the CPU activity which caused the system to blow-up, i.e., the resource limiting activity, was examined in each of the six model sets. The following conclusions were reached:

(1) Message Switch MP - message exchange and code conversion

(2) Base Switch MP - message exchange and code conversion

(3) Packet Switch MP - sheer character manipulation

(4) Message Switch FE - character examination and conversion

(5) Base Switch FE - character examination and conversion

(6) Packet Switch FE - sheer character handling and examination

SUMMARY OF RUN RESULTS FOR PACKET SWITCH, FRONT-END PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC) 2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
Typical System		6100-6103	.2223	180.68%	.0035	.3415
Best System		6200-6203	.3456	336.40%	.0030	.4508
Worst System		6300-6303	.0792	0.00%	.0066	.2300
# of Register Sets						
1 (Worst)		6300-6303	.0792	0.00%	.0066	.2300
2		6250-6253	.1071	35.20%	.0060	.2790
3		6260-6263	.1071	35.20%	.0060	.2791
Shift		6700-6703	.0906	14.40%	.0059	.2385
Index/Indirect						
8 Registers No Indirect		6780-6783	.1066	34.60%	.0056	.2580
4 Registers No Indirect		6785-6788	.0977	23.40%	.0059	.2488
0 Registers No Indirect		6300-6303	.0792	0.00%	.0066	.2300
8 Registers With Indirect		6750-6753	.1258	58.80%	.0052	.2803
4 Registers With Indirect		6760-6763	.1147	44.80%	.0055	.2696

TABLE III-17

SUMMARY OF RUN RESULTS FOR PACKET SWITCH, FRONT-END PROCESSOR

CASE	COMMENTS	RUN #'s	CAPACITY (MSG/SEC)2	% IMPROVEMENT OVER "WORST" SYSTEM	ZERO TRAFFIC CYCLE LENGTH (SECS.)	INPUT TRAFFIC AT BLOWUP POINT (MSG/SEC)
O Registers With Indirect		6775-6778	.0940	18.70%	.0062	.2485
Memory Speed						
.01	Infinitely Fast Memory	6940-6943	795.4700		.0000657	23.0016
.25	Best Likely Memory	6950-6953	1.3470	1600.00%	.0016	.92006
.50	Most Likely Memory	6840-6844	.3368	325.00%	.0033	.4600
1.00	Standard Memory	6300-6303	.0792	0.00%	.0066	.2300
2.00	Worst Possible Memory	6890-6894	.0206	- 74.00%	.0131	.1150

TABLE III-17 (Continued)

	Message Switch	Base Switch	Packet Switch
MP Delay	2.000	5.000	.200
FE Delay	.100	.100	.050
Input Rate (MP)	2.062	1.375	1.760
Input Rate (FE)	.750	.510	.300
Needed FE	2.750	2.700	5.870
MP Memory	7.000	17.000	2.000
FE Memory	2.750	2.700	5.870
Total Memory	9.750	16.700	7.870
System Weight	.140	.430	.430

Average Memory = 11.93

TABLE III-18

TABLE III-19

System	Message Switch MP	Message Switch FE	Base Switch MP	Base Switch FE	Packet Switch MP	Packet Switch FE
Unit Weight	1.00	2.75	1.00	2.70	1.00	5.87
System Weight	.14	.14	.43	.43	.43	.43
Net Weight	.14	.39	.43	1.16	.43	2.52
Normalized Weight	.03	.08	.08	.23	.08	.50

Feature	Message Switch MP UNW	Message Switch MP W	Message Switch FE UNW	Message Switch FE W	Base Switch MP UNW	Base Switch MP W	Base Switch FE UNW	Base Switch FE W	Packet Switch MP UNW	Packet Switch MP W	Packet Switch FE UNW	Packet Switch FE W	Average
Typical System	142.00	4.2600	215.80	17.2600	165.00	13.200	228.40	52.530	528.00	42.240	180.68	90.34	220.00%
Best System	150.00	4.5000	344.00	27.5200	170.00	13.600	369.00	84.870	593.00	47.440	336.40	168.20	346.00%
2 Reg. Sets	.71	.0213	21.10	1.6880	.49	.040	20.50	4.715	9.78	.782	35.20	17.60	24.90%
3 Reg. Sets	1.20	.0360	21.10	1.6880	.82	.065	20.60	4.738	11.15	.892	35.20	17.60	25.00%
Paging	15.30	.4590			18.50	1.480			47.66	3.813			5.75%
Move Inst.	.53	.0159			.24	.019			0.00	0.000			.04%
Clear Mem.	.77	.0231			.50	.040			7.93	.634			.70%
Link List	.07	.0021			.05	.004			0.00	0.000			.01%

Shift	UNW	W	UNW	W	UNW	W	UNW	W	UNW	W	UNW	W	%
8 Reg./No	15.40	.4620	124.20	1.1360	22.50	1.800	14.40	3.312	33.34	2.667	14.40	7.20	16.60%
4 Reg./No	26.70	.8010	40.26	3.2210	43.30	3.464	41.59	9.566	66.01	5.281	34.60	17.30	39.60%
8 Reg./Yes	15.00	.4500	25.82	2.0660	30.30	2.420	26.67	6.134	34.30	2.744	23.40	11.70	25.50%
4 Reg./Yes	36.70	1.1010	60.14	4.8110	45.90	3.670	63.24	14.540	75.41	6.033	58.80	29.40	59.60%
0 Reg./Yes	25.00	.7500	43.67	3.4940	32.50	2.600	45.47	10.460	41.38	3.310	44.80	22.40	43.00%
250 Nano	2.00	.0600	12.07	.9656	1.60	.128	12.23	2.813	12.82	1.026	18.70	9.35	14.30%
800 Nano	357.00	10.7100	1429.00	114.3200	533.00	42.640	1522.00	350.060	1356.00	108.480	1600.00	800.00	142.00%
2 Micro-Seconds	134.00	4.0200	282.60	22.6100	183.00	14.640	305.70	70.310	211.70	23.280	325.00	163.00	297.00%
	-70.00	-2.1000	-76.00	-6.0800	-66.00	-5.280	-75.00	-17.250	-75.00	-6.00	-74.00	37.00	-73.70%

UNW = Unweighted
W = Weighted

TABLE III-19 (Continued)

What it came down to in all cases was that there comes a point where the sheer character manipulation and examination dominated the system load. For this reason, the following features are required:

- Translate instructions

- Translate and test instruction

- Character indexing

- Sequential jump table instructions

9.1.2 Peripherals

Since in no case were the peripherals much of a problem, there is no need to make a special effort in this area. Commercially available disc units should be adequate to most systems.

10.0 CONCLUSIONS AND RECOMMENDATIONS

On the basis of the above modeling effort, the following conclusions and recommendations have been reached.

(1) Total Amount of Memory

Should be potentially large. The typical configuration will require 12 modules of 65,536 characters each.

(2) Number of Register Sets

Provisions should be made to allow more than one set of registers (at least two) to reduce interrupt related overhead. This must be a modular feature as many configurations will be adequate with only one register set.

(3) Paging

Some form of paging is desirable in order to simplify the mechanization of adequate memory protection. However, page size should be kept large - of the order of 65K. If the page size is made equal to physical memory module size, considerable simplification in the design of the system will be obtained.

(4) Number of Registers

The system should have 16 word sized (32 bit) general purpose registers.

(5) Indirect Addressing

A single level of indirect addressing should be provided.

(6) Shift Instructions

Adequate shift instructions should be provided.

(7) Move Instruction

Should be provided only if done at very low additional or no additional cost.

(8) Clear Memory Instruction

Should be provided only if done at no or very little additional cost.

(9) Link List Instruction

This goes somewhat further than the simple question of the link list instruction. It also affects the various addressing modes that were considered in detail during the study. If the block index mode had been useful, it would have shown up in this instruction. On the basis of this, the elimination of the various block address and block index modes is recommended as well as the link list instruction.

(10) Memory Speed

Speed should be established at the cost-optimum point for the time period of the deployment - this will probably be in the 400 to 600 nanosecond region. The components of the entire CPU must be speed independent (anisochronous design) to assure tracking the cost-optimum point as the memory speeds improve.

(11) Additional Features

Careful consideration must be given to the employment of character translation and test instructions for use in the front-ends.

(12) Peripheral Memories

The main stream peripheral memories (e.g., disc) that are commercially available should be adequate to the majority of sites. No special effort should be made to develop special memories. Any proposed memory

III-79

(e.g., bubble domain) will have to make it on its
own, on the basis of cost, rather than cost-perfor-
mance.

11.0 COMMUNICATIONS COMPUTER COMPLEX (CCC) MODELING

11.1 General

The modeling effort described above was used to determine
the desirable and necessary characteristics of a CPU to be used
in a CPS environment. The models were based on "classical"
existing system architectures.

During the final phase of this project, the CPS Study
Program, a Communication Computer Complex (CCC) architecture was
developed which satisfied the many diverse requirements itemized
in Volume II, Sections 2 and 3 of this report. The CCC architec-
ture, which is described in detail in Volume IV of the report, is
unique in many aspects. For this reason, several modeling runs
were performed to determine the relative merits of the CCC archi-
tecture as compared to the "classical" architectures.

11.2 Conclusions

To the extent that it was possible to model the new archi-
tecture, it showed a 30% improvement over the "best" system pre-
viously examined and a 40% improvement over the "typical" system
examined. These comparisons were done at a memory cycle time of
1 microsecond. Increase in memory speed to a projected 500 nano-
seconds would more than double the predicted performance. The
nature of the modeling is such that these improvements represent
a lower bound. Actual improvement is expected to be significantly
better.

11.3 Methodology

The previous modeling effort involved the examination of
six sets of models - three message processing and three front-
end processing models, for each of message switching, packet
switching and base distribution switching. In each of these
models numerous questions relating to the architecture were in-
vestigated. The modeling discussed here was more restricted in
scope, intended primarily to provide quantitative indications of
performance improvements due to the proposed architecture. There-
fore, only one model was examined - the message processing mes-
sage switching system model.

Traffic and system tuning parameters were not modified.
The latter, particularly, tends to make the results somewhat
pessimistic since we did not take advantage of the new architec-
ture to tune the model system to a higher performance level.
Hardware parameter values were changed to the extent possible,

to reflect improvements in the repertoire. The basic set of parameters are shown in Table III-20. All other parameter values were identical to those previously used. A further set of modifications were made to subroutine and macro models to reflect the fact that some of these functions would be obviated or handled in the ICU rather than in the MP CPU. These values are also shown in Table III-20.

The model therefore does not take into account the different coding approaches that might be taken with the new architecture. It is essentially a software model which has been adapted (one-for-one) and cannot therefore, take advantage of all the features available. Consequently, it is expected that actual performance will be significantly better than predicted by these runs.

11.4 Run Results

11.4.1 Basic Runs

The basic set of runs are shown in Tables III-21 through III-25. A plot of MP program cycle length is a function of input traffic is shown as Figure III-4. Figure III-5 shows the subjective rate vs. input traffic for the same situation. This is the curve whose integral is defined as the system capacity. Figure III-6 shows the cycle length, subjective rate and total rates, superimposed for comparison.

11.4.2 Impact of Memory Speed

Figure III-7 shows the impact on cycle length of varying the memory speed over a range of 100 nanoseconds to 1 microsecond. The improvement factors does not materially differ from the kind of improvements investigated and described above. CPU binding is still the ultimate limitation.

11.4.3 Expected Impact on Other Models

It is expected that there will be a slightly better improvement in the front-end and in the packet switch models. Possibly bringing the effective capacity improvement to 50% overall not counting the effect of higher memory speeds.

A	=	1.6666660		VLUT	=	1.0000000
B	=	1.1666667		VMDL	=	1.0000000
C	=	2.1666666		VMOV	=	0.0000000
D	=	2.1666666		VODB	=	0.0000000
DNIP	=	4.0000000		VODO	=	0.0000000
E	=	1.0000000		VODU	=	0.0000000
F	=	0.9000000		VODX	=	1.0000000
G	=	0.0000000		VPSA	=	1.0000000
H	=	2.0000000		VPKT	=	0.0000000
I	=	2.0000000		VPRI	=	0.0000000
INBL	=	0.0000000		VROW	=	0.0000000
INPK	=	0.0000000		VSST	=	1.0000000
INSS	=	0.0000000		VSIP	=	1.0000000
J	=	2.0000000		VSLI	=	0.0000000
JBT	=	0.0020000		VTDJ	=	1.0000000
JCH	=	17.4999998		VTDR	=	1.0000000
JSC	=	0.0015000		VTLS	=	0.0000000
JSL	=	1.5000000		VTPH	=	0.0000000
JSQ	=	1.5000000		VTPS	=	1.0000000
JTR	=	4.0000000		VZRO	=	0.0000000
K	=	0.9000000		ZNRS	=	4.0000000
S26	=	32.6299996				
S43	=	4.4999999				
S48	=	2.3333330				
S49	=	8.6300000				
S57	=	0.0000000				
S58	=	0.0000000				
U10	=	0.0000000				
U11	=	0.0000000				
U12	=	0.0000000				
U13	=	0.0000000				
U14	=	0.0000000				
U16	=	0.0000000				
U17	=	0.0000000				
U2	=	2.0000000				
U6	=	3.0000000				
U8	=	39.7299995				
VBIN	=	1.0000000				
VBNK	=	1.0000000				
VCHA	=	0.0000000				
VCIO	=	1.0000000				
VCOM	=	0.0000000				
VCSI	=	0.0000000				
VFE	=	0.0000000				
VHSS	=	1.0000000				
VITS	=	1.5000000				
VLED	=	1.5000000				

TABLE III-20

PARAMETER FILE VALUES USED IN SETTING UP CONFIRMATION RUN MODELS

CCC ARCHITECTURE PERFORMANCE CONFIRMATION RUNS

IMPACT OF MEMORY SPEED ON MESSAGE SWITCH
MESSAGE PROCESSOR - MEMORY CYCLE = 1.0 MICROSECONDS

FOR:

FUNCTION FILE: VER6
PARAMETER FILE: RUNS

AND:

| VARIABLE | | RANGE OF VALUES | | |
NAME	LEVEL	FROM	TO	IN STEPS OF
TRFI	1	0.100E-05	5.10000	0.10000

****

LINE NO.	OBJT RATE	SUBJ RATE	MEMR MODS	CYCL LENT	TOTL RATE
1	0.100E-05	2.70166	6.00000	0.24165	2.70166
2	0.10000	2.64133	6.00000	0.24261	2.74133
3	0.20000	2.43888	6.00000	0.25512	2.63888
4	0.30000	2.24559	6.00000	0.26874	2.54559
5	0.40000	2.06331	6.00000	0.28532	2.46331
6	0.50000	1.91716	6.00000	0.30102	2.41716
7	0.60000	1.79961	6.00000	0.31565	2.39961
8	0.70000	1.70480	6.00000	0.32907	2.40480
9	0.80000	1.62675	6.00000	0.34143	2.42675
10	0.90000	1.56304	7.00000	0.35261	2.46304
11	1.00000	1.50984	7.00000	0.36282	2.50984
12	1.10000	1.46461	7.00000	0.37222	2.56461
13	1.20000	1.42589	7.00000	0.38089	2.62590
14	1.30000	1.39210	7.00000	0.38897	2.69210
15	1.40000	1.36213	7.00000	0.39660	2.76213
16	1.50000	1.33541	7.00000	0.40381	2.83542
17	1.60000	1.31100	7.00000	0.41075	2.91100
18	1.70000	1.28870	7.00000	0.41742	2.98870
19	1.80000	1.26793	7.00000	0.42392	3.06793
20	1.90000	1.24855	7.00000	0.43027	3.14855
21	2.00000	1.23025	7.00000	0.43651	3.23025
22	2.10000	1.17146	7.00000	0.45503	3.27146
23	2.20000	1.10389	7.00000	0.47852	3.30389
24	2.30000	1.03724	7.00000	0.50469	3.33724
25	2.40000	0.97152	7.00000	0.53401	3.37152

TABLE III-21

CCC ARCHITECTURE PERFORMANCE CONFIRMATION RUNS

RUN NO. 9000   JANUARY 16, 1976   PAGE 2 of 2

LINE NO.	OBJT RATE	SUBJ RATE	MEMR MODS	CYCL LENT	TOTL RATE
26	2.50000	0.90696	7.00000	0.56694	3.40696
27	2.60000	0.84507	7.00000	0.60320	3.44507
28	2.70000	0.78397	7.00000	0.64460	3.48397
29	2.80000	0.72363	8.00000	0.69235	3.52363
30	2.90000	0.66400	8.00000	0.74807	3.56400
31	3.00000	0.60504	8.00000	0.81395	3.60504
32	3.10000	0.54425	8.00000	0.89716	3.64425
33	3.20000	0.47996	8.00000	1.00869	3.67996
34	3.30000	0.41709	8.00000	1.15085	3.71709
35	3.40000	0.37843	9.00000	1.25757	3.77843
36	3.50000	0.35777	9.00000	1.32196	3.85777
37	3.60000	0.33732	9.00000	1.39345	3.93732
38	3.70000	0.31708	9.00000	1.47323	4.01708
39	3.80000	0.29725	9.00000	1.56173	4.09725
40	3.90000	0.27760	10.00000	1.66194	4.17760
41	4.00000	0.25809	10.00000	1.77646	4.25809
42	4.10000	0.23888	10.00000	1.90738	4.33·88
43	4.20000	0.21980	11.00000	2.06000	4.41980
44	4.30000	0.20087	11.00000	2.23990	4.50087
45	4.40000	0.18203	12.00000	2.45621	4.58203
46	4.50000	0.16325	12.00000	2.72129	4.66325
47	4.60000	0.14440	13.00000	3.05680	4.74440
48	4.70000	0.12532	14.00000	3.49906	4.82532
49	4.80000	0.10468	15.00000	4.16104	4.90468
50	4.90000	0.07896	18.00000	5.47659	4.97896
51	5.00000	***	***	***	***

TABLE III-21 (Continued)

CCC ARCHITECTURE PERFORMANCE CONFIRMATION RUNS

RUN NO. 9001   JANUARY 16, 1976   PAGE 1 OF 2

IMPACT OF MEMORY SPEED ON MESSAGE SWITCH
MESSAGE PROCESSOR - MEMORY CYCLE = 1.0 MICROSECONDS

FOR:

FUNCTION FILE: VER6
PARAMETER FILE: RUNS

AND:

VARIABLE NAME	LEVEL	RANGE OF VALUES FROM	TO	IN STEPS OF
TRFI	1	0.100E-05	5.10000	0.10000

****

LINE NO.	INPU MPS	OUTP MPS	THRU MPS	CYCL LENT	MSG DELY
1	0.100E-05	0.150E-05	0.250E-05	0.24165	0.37014
2	0.10000	0.15036	0.25036	0.24261	0.37860
3	0.20000	0.30073	0.50073	0.25512	0.41003
4	0.30000	0.45109	0.75109	0.26874	0.44532
5	0.40000	0.60145	1.00145	0.28532	0.48466
6	0.50000	0.75181	1.25181	0.30102	0.52161
7	0.60000	0.90217	1.50217	0.31565	0.55568
8	0.70000	1.05254	1.75254	0.32907	0.58658
9	0.80000	1.20290	2.00290	0.34143	0.61472
10	0.90000	1.35326	2.25326	0.35261	0.63978
11	1.00000	1.50362	2.50362	0.36282	0.66232
12	1.10000	1.65398	2.75398	0.37222	0.68277
13	1.20000	1.80434	3.00435	0.38089	0.70131
14	1.30000	1.95471	3.25471	0.38897	0.71834
15	1.40000	2.10507	3.50507	0.39660	0.73415
16	1.50000	2.25543	3.75543	0.40381	0.74883
17	1.60000	2.40579	4.00579	0.41075	0.76278
18	1.70000	2.55615	4.25616	0.41742	0.77598
19	1.80000	2.70652	4.50652	0.42392	0.78869
20	1.90000	2.85688	4.75688	0.43027	0.80093
21	2.00000	3.00724	5.00724	0.43651	0.81285
22	2.10000	3.15760	5.25760	0.45503	0.85364
23	2.20000	3.30796	5.50796	0.47852	0.90589
24	2.30000	3.45833	5.75833	0.50469	0.96410
25	2.40000	3.60869	6.00869	0.53401	1.02931

TABLE III-22

LINE NO.	INPU MPS	OUTP MPS	THRU MPS	CYCL LENT	MSG DELY
26	2.50000	3.75905	6.25905	0.56694	1.10258
27	2.60000	3.90941	6.50941	0.60320	1.18333
28	2.70000	4.05977	6.75977	0.64460	1.27556
29	2.80000	4.21014	7.01014	0.69235	1.38193
30	2.90000	4.36050	7.26050	0.74807	1.50603
31	3.00000	4.51086	7.51086	0.81395	1.65277
32	3.10000	4.66122	7.76122	0.89716	1.83738
33	3.20000	4.81158	8.01158	1.00869	2.08351
34	3.30000	4.96195	8.26195	1.15085	2.39756
35	3.40000	5.11231	8.51231	1.25757	2.64249
36	3.50000	5.26267	8.76267	1.32196	2.79509
37	3.60000	5.41303	9.01303	1.39345	2.96457
38	3.70000	5.56339	9.26339	1.47323	3.15381
39	3.80000	5.71376	9.51376	1.56173	3.36414
40	3.90000	5.86412	9.76412	1.66194	3.60235
41	4.00000	6.01448	10.01448	1.77646	3.87464
42	4.10000	6.16484	10.26484	1.90738	4.18627
43	4.20000	6.31520	10.51520	2.06000	4.54968
44	4.30000	6.46556	10.76557	2.23990	4.97825
45	4.40000	6.61593	11.01593	2.45621	5.49367
46	4.50000	6.76629	11.26629	2.72129	6.12553
47	4.60000	6.91665	11.51665	3.05680	6.92532
48	4.70000	7.06701	11.76701	3.49906	7.97951
49	4.80000	7.21737	12.01738	4.16104	9.55332
50	4.90000	7,36774	12.26774	5.47659	12.66494
51	5.00000	7.51810	12.51810	***	***

TABLE III-22 (Continued)

CCC ARCHITECTURE PERFORMANCE CONFIRMATION RUNS

RUN NO. 9002   JANUARY 16, 1976   PAGE 1 OF 2

IMPACT OF MEMORY SPEED ON MESSAGE SWITCH
MESSAGE PROCESSOR - MEMORY CYCLE = 1.0 MICROSECONDS

FOR:

FUNCTION FILE: VER6
PARAMETER FILE: RUNS

AND:

| VARIABLE | | RANGE OF VALUES | | |
NAME	LEVEL	FROM	TO	IN STEPS OF
TRFI	1	0.100E-05	5.10000	0.10000

****

LINE NO.	INPU CPS	OUTP CPS	THRU CPS	CYCL LENT	MSG DELY
1	0.00885	0.314E-03	0.01870	0.24165	0.37014
2	885.01215	31.42831	1870.50	0.24261	0.37860
3	1770.02	62.85631	3740.99	0.25512	0.41003
4	2655.01	94.28431	5611.47	0.26874	0.44532
5	3540.02	125.71231	7481.96	0.28532	0.48466
6	4425.01	157.14030	9352.43	0.30102	0.52161
7	5310.02	188.56830	11222.93	0.31565	0.55568
8	6195.03	219.99630	13093.41	0.32907	0.58658
9	7080.03	251.42430	14963.90	0.34143	0.61472
10	7965.04	282.85229	16834.39	0.35261	0.63978
11	8850.03	314.28029	18704.87	0.36282	0.66232
12	9735.03	345.70829	20575.35	0.37222	0.68277
13	10620.03	377.13629	22445.82	0.38089	0.70131
14	11505.03	408.56429	24316.31	0.38897	0.71834
15	12390.04	439.99229	26186.80	0.39660	0.73415
16	13275.07	471.42029	28057.31	0.40381	0.74883
17	14160.05	502.84828	29927.77	0.41075	0.76278
18	15045.06	534.27628	31798.27	0.41742	0.77598
19	15930.06	565.70428	33668.75	0.42392	0.78869
20	16815.05	597.13228	35539.23	0.43027	0.80093
21	17700.05	628.56028	37409.71	0.43651	0.81285
22	18585.06	659.98827	39280.20	0.45503	0.85364
23	19470.04	691.41627	41150.66	0.47852	0.90589
24	20355.07	722.84427	43021.18	0.50469	0.96410
25	21240.05	754.27228	44891.64	0.53401	1.02931

TABLE III-23

CCC ARCHITECTURE PERFORMANCE CONFIRMATION RUNS

RUN NO. 9002   JANUARY 16, 1976   PAGE 2 OF 2

LINE NO.	INPU CPS	OUTP CPS	THRU CPS	CYCL LENT	MSG DELY
26	22125.05	785.70028	46762.12	0.56694	1.10258
27	23010.06	817.12827	48632.61	0.60320	1.18333
28	23895.06	848.55627	50503.09	0.64460	1.27556
29	24780.06	879.98427	52373.57	0.69235	1.38193
30	25665.10	911.41226	54244.10	0.74807	1.50603
31	26550.12	942.84026	56114.60	0.81395	1.65277
32	27435.13	974.26824	57985.09	0.89716	1.83738
33	28320.08	1005.70	59855.52	1.00869	2.08351
34	29205.12	1037.12	61726.05	1.15085	2.39756
35	30090.10	1068.55	63596.51	1.25757	2.64249
36	30975.12	1099.98	65467.01	1.32196	2.79509
37	31860.10	1131.41	67337.47	1.39345	2.96457
38	32745.14	1162.84	69208.00	1.47323	3.15381
39	33630.09	1194.26	71078.43	1.56173	3.36414
40	34515.10	1225.69	72948.92	1.66194	3.60235
41	35400.12	1257.12	74819.42	1.77646	3.87464
42	36285.16	1288.55	76689.95	1.90738	4.18627
43	37170.11	1319.98	78560.38	2.06000	4.54968
44	38055.06	1351.40	80430.81	2.23990	4.97825
45	38940.10	1382.83	82301.33	2.45621	5.49367
46	39825.12	1414.26	84171.83	2.72129	6.12553
47	40710.13	1445.69	86042.33	3.05680	6.92532
48	41595.07	1477.12	87912.75	3.49906	7.97951
49	42480.09	1508.54	89783.25	4.16104	9.55332
50	43365.14	1539.97	91653.78	5.47659	12.66494
51	***	1571.40	***	***	***

TABLE III-23 (Continued)

CCC ARCHITECTURE PERFORMANCE CONFIRMATION RUNS

RUN NO. 9003   JANUARY 16, 1976   PAGE 1 OF 2

IMPACT OF MEMORY SPEED ON MESSAGE SWITCH
MESSAGE PROCESSOR - MEMORY CYCLE = 1.0 MICROSECONDS

FOR:

FUNCTION FILE: VER6
PARAMETER FILE: RUNS

AND:

| VARIABLE | | RANGE OF VALUES | | |
NAME	LEVEL	FROM	TO	IN STEPS OF
TRFI	1	0.100E-05	5.10000	0.10000

****

LINE NO.	DATA CHAR	QUEU CHAR	JOUR CHAR	FIXD CHAR	TOTA CHAR
1	2961.11	71.68956	1.98802	367680.00	370714.79
2	3250.01	2691.44	10.52379	367680.00	373631.97
3	3644.17	4630.00	16.51955	367680.00	375970.68
4	4160.91	6466.03	23.64590	367680.00	378330.58
5	4789.78	8249.41	32.29263	367680.00	380751.48
6	5503.83	9999.20	40.87224	367680.00	383223.90
7	6285.17	11725.21	50.47792	367680.00	385740.86
8	7117.19	13433.21	60.57443	367680.00	388290.97
9	7990.97	15126.99	71.09413	367680.00	390869.05
10	8891.20	16809.13	81.86508	367680.00	393462.18
11	9813.96	18481.53	92.86437	367680.00	396068.34
12	10756.33	20145.64	104.07300	367680.00	398686.04
13	11713.46	21802.56	115.44085	367680.00	401311.46
14	12685.33	23453.19	126.97890	367680.00	403945.49
15	13672.14	25098.26	138.69840	367680.00	406589.10
16	14671.46	26738.36	150.57316	367680.00	409240.39
17	15687.36	28374.00	162.66251	367680.00	411904.02
18	16717.23	30005.60	174.93539	367680.00	414577.76
19	17764.45	31633.54	187.43991	367680.00	417265.43
20	18828.02	33258.12	200.16515	367680.00	419966.30
21	19909.96	34879.63	213.14003	367680.00	422682.72
22	21795.99	36501.95	236.63149	367680.00	426214.57
23	24094.09	38123.54	265.47376	367680.00	430163.10
24	26681.05	39744.05	298.04604	367680.00	434403.14
25	29607.62	41363.91	335.00588	367680.00	438986.52

TABLE III-24

CCC ARCHITECTURE PERFORMANCE CONFIRMATION RUNS

LINE NO.	DATA CHAR	QUEU CHAR	JOUR CHAR	FIXD CHAR	TOTA CHAR
26	32927.78	42983.55	377.04871	367680.00	443968.37
27	36641.94	44603.16	424.15211	367680.00	449349.24
28	40912.16	46223.67	478.43473	367680.00	455294.26
29	45865.64	47845.87	541.53722	367680.00	461933.04
30	51672.29	49470.77	615.65083	367680.00	469438.71
31	58564.60	51099.70	703.77493	367680.00	478048.07
32	67139.63	52736.24	813.39895	367680.00	488369.27
33	78366.51	54386.04	956.76009	367680.00	501389.30
34	92741.31	56051.35	1140.59	367680.00	517613.25
35	105281.31	57702.41	1301.27	367680.00	531964.98
36	114251.88	59334.50	1415.24	367680.00	542681.61
37	124242.46	60970.56	1542.27	367680.00	554435.27
38	135429.00	62611.54	1684.63	367680.00	567405.16
39	147945.88	64258.04	1844.15	367680.00	581728.06
40	162146.64	65912.07	2025.26	367680.00	597763.95
41	178398.26	67575.57	2232.68	367680.00	615886.49
42	197067.29	69250.33	2471.19	367680.00	636468.80
43	218854.37	70940.13	2749.89	367680.00	660224.38
44	244576.17	72649.15	3079.42	367680.00	687984.73
45	275503.83	74384.04	3475.94	367680.00	721043.80
46	313406.61	76153.93	3962.24	367680.00	761202.77
47	361285.22	77974.48	4576.92	367680.00	811516.61
48	424185.39	79871.92	5384.91	367680.00	877122.22
49	516804.73	81924.88	6573.95	367680.00	972983.54
50	695052.33	84433.24	8858.09	367680.00	0.116E+07
51	***	***	***	367680.00	***

TABLE III-24 (Continued)

CCC ARCHITECTURE PERFORMANCE CONFIRMATION RUNS

RUN NO. 9004   JANUARY 16, 1976   PAGE 1 OF 2

IMPACT OF MEMORY SPEED ON MESSAGE SWITCH
MESSAGE PROCESSOR – MEMORY CYCLE = 1.0 MICROSECONDS
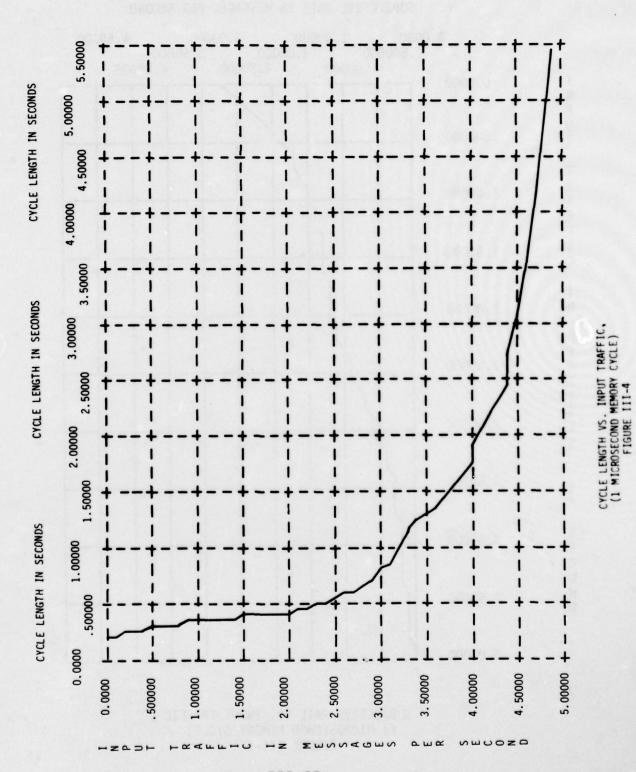
FOR:

FUNCTION FILE: VER6
PARAMETER FILE: RUNS

AND:

VARIABLE		RANGE OF VALUES		
NAME	LEVEL	FROM	TO	IN STEPS OF
TRFI	1	0.100E-05	5.10000	0.10000

****

LINE NO.	DATA BLKS	QUEU BLKS	PRGM CHAR	TABL CHAR	MEMR MODS
1	59.00000	13.00000	140000.00	203680.00	6.00000
2	63.00000	177.00000	140000.00	203680.00	6.00000
3	69.00000	298.00000	140000.00	203680.00	6.00000
4	78.00000	413.00000	140000.00	203680.00	6.00000
5	87.00000	524.00000	140000.00	203680.00	6.00000
6	98.00000	633.00000	140000.00	203680.00	6.00000
7	111.00000	741.00000	140000.00	203680.00	6.00000
8	124.00000	848.00000	140000.00	203680.00	6.00000
9	137.00000	954.00000	140000.00	203680.00	6.00000
10	151.00000	1059.00	140000.00	203680.00	7.00000
11	166.00000	1164.00	140000.00	203680.00	7.00000
12	181.00000	1268.00	140000.00	203680.00	7.00000
13	196.00000	1371.00	140000.00	203680.00	7.00000
14	211.00000	1474.00	140000.00	203680.00	7.00000
15	226.00000	1577.00	140000.00	203680.00	7.00000
16	242.00000	1680.00	140000.00	203680.00	7.00000
17	258.00000	1782.00	140000.00	203680.00	7.00000
18	274.00000	1884.00	140000.00	203680.00	7.00000
19	290.00000	1986.00	140000.00	203680.00	7.00000
20	307.00000	2087.00	140000.00	203680.00	7.00000
21	324.00000	2188.00	140000.00	203680.00	7.00000
22	353.00000	2290.00	140000.00	203680.00	7.00000
23	389.00000	2391.00	140000.00	203680.00	7.00000
24	429.00000	2493.00	140000.00	203680.00	7.00000
25	475.00000	2594.00	140000.00	203680.00	7.00000

TABLE III-25

LINE NO.	DATA BLKS	QUEU BLKS	PRGM CHAR	TABL CHAR	MEMR MODS
26	527.00000	2695.00	140000.00	203680.00	7.00000
27	585.00000	2796.00	140000.00	203680.00	7.00000
28	652.00000	2897.00	140000.00	203680.00	7.00000
29	729.00000	2999.00	140000.00	203680.00	8.00000
30	820.00000	3100.00	140000.00	203680.00	8.00000
31	928.00000	3202.00	140000.00	203680.00	8.00000
32	1062.00	3305.00	140000.00	203680.00	8.00000
33	1237.00	3408.00	140000.00	203680.00	8.00000
34	1462.00	3512.00	140000.00	203680.00	8.00000
35	1658.00	3615.00	140000.00	203680.00	9.00000
36	1798.00	3717.00	140000.00	203680.00	9.00000
37	1954.00	3819.00	140000.00	203680.00	9.00000
38	2129.00	3922.00	140000.00	203680.00	9.00000
39	2324.00	4025.00	140000.00	203680.00	9.00000
40	2546.00	4128.00	140000.00	203680.00	10.00000
41	2800.00	4232.00	140000.00	203680.00	10.00000
42	3092.00	4337.00	140000.00	203680.00	10.00000
43	3432.00	4442.00	140000.00	203680.00	11.00000
44	3834.00	4549.00	140000.00	203680.00	11.00000
45	4317.00	4658.00	140000.00	203680.00	12.00000
46	4909.00	4768.00	140000.00	203680.00	12.00000
47	5658.00	4882.00	140000.00	203680.00	13.00000
48	6640.00	5000.00	140000.00	203680.00	14.00000
49	8088.00	5129.00	140000.00	203680.00	15.00000
50	10873.00	5286.00	140000.00	203680.00	18.00000
51	***	***	140000.00	203680.00	***

TABLE III-25 (Continued)

CYCLE LENGTH VS. INPUT TRAFFIC.
(1 MICROSECOND MEMORY CYCLE)
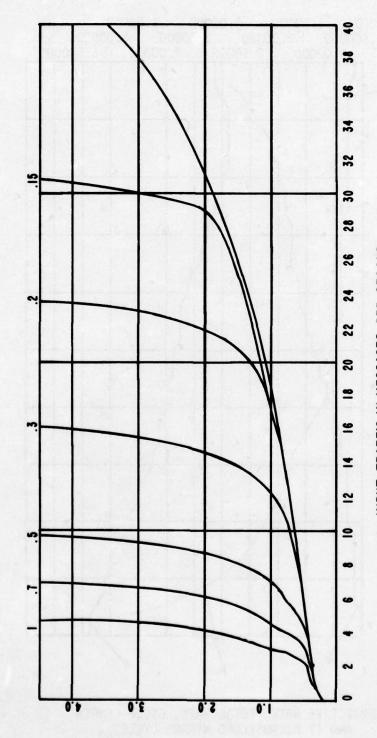FIGURE III-4
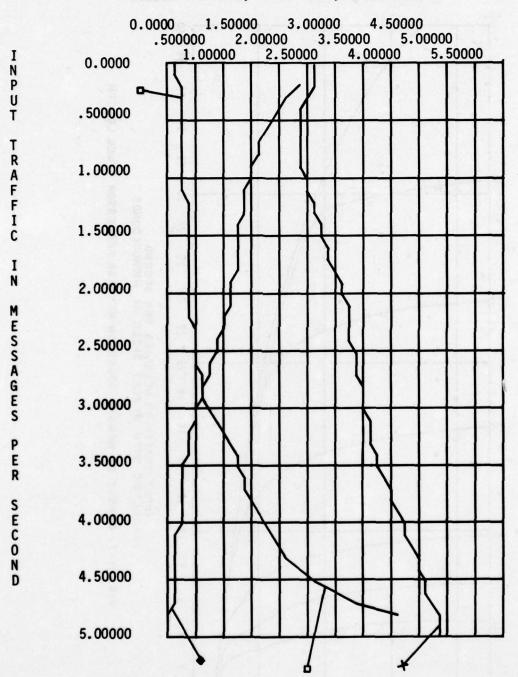
SUBJECTIVE RATE VS. INPUT TRAFFIC
(1 MICROSECOND MEMORY CYCLE)
FIGURE III-5

III-94

SUBJECTIVE RATE, TOTAL RATE, CYCLE LENGTH

SUBJECTIVE RATE, TOTAL RATE, CYCLE LENGTH
AND (1 MICROSECOND MEMORY CYCLE)
FIGURE III-6

INPUT TRAFFIC IN MESSAGES PER SECOND
PARAMETRIC WITH MEMORY SPEED IN MICROSECONDS

FIG. III-7 - EFFECT OF MEMORY SPEED ON MESSAGE PROCESSOR CYCLE LENGTH

METRIC SYSTEM

BASE UNITS:

Quantity	Unit	SI Symbol	Formula
length	metre	m	...
mass	kilogram	kg	...
time	second	s	...
electric current	ampere	A	...
thermodynamic temperature	kelvin	K	...
amount of substance	mole	mol	...
luminous intensity	candela	cd	...

SUPPLEMENTARY UNITS:

plane angle	radian	rad	...
solid angle	steradian	sr	...

DERIVED UNITS:

Acceleration	metre per second squared	...	m/s
activity (of a radioactive source)	disintegration per second	...	(disintegration)/s
angular acceleration	radian per second squared	...	rad/s
angular velocity	radian per second	...	rad/s
area	square metre	...	m
density	kilogram per cubic metre	...	kg/m
electric capacitance	farad	F	A·s/V
electrical conductance	siemens	S	A/V
electric field strength	volt per metre		V/m
electric inductance	henry	H	V·s/A
electric potential difference	volt	V	W/A
electric resistance	ohm		V/A
electromotive force	volt	V	W/A
energy	joule	J	N·m
entropy	joule per kelvin	...	J/K
force	newton	N	kg·m/s
frequency	hertz	Hz	(cycle)/s
illuminance	lux	lx	lm/m
luminance	candela per square metre	...	cd/m
luminous flux	lumen	lm	cd·sr
magnetic field strength	ampere per metre	...	A/m
magnetic flux	weber	Wb	V·s
magnetic flux density	tesla	T	Wb/m
magnetomotive force	ampere	A	...
power	watt	W	J/s
pressure	pascal	Pa	N/m
quantity of electricity	coulomb	C	A·s
quantity of heat	joule	J	N·m
radiant intensity	watt per steradian	...	W/sr
specific heat	joule per kilogram-kelvin	...	J/kg·K
stress	pascal	Pa	N/m
thermal conductivity	watt per metre-kelvin	...	W/m·K
velocity	metre per second	...	m/s
viscosity, dynamic	pascal-second	...	Pa·s
viscosity, kinematic	square metre per second	...	m/s
voltage	volt	V	W/A
volume	cubic metre	...	m
wavenumber	reciprocal metre	...	(wave)/m
work	joule	J	N·m

SI PREFIXES:

Multiplication Factors	Prefix	SI Symbol
$1\ 000\ 000\ 000\ 000 = 10^{12}$	tera	T
$1\ 000\ 000\ 000 = 10^{9}$	giga	G
$1\ 000\ 000 = 10^{6}$	mega	M
$1\ 000 = 10^{3}$	kilo	k
$100 = 10^{2}$	hecto*	h
$10 = 10^{1}$	deka*	da
$0.1 = 10^{-1}$	deci*	d
$0.01 = 10^{-2}$	centi*	c
$0.001 = 10^{-3}$	milli	m
$0.000\ 001 = 10^{-6}$	micro	μ
$0.000\ 000\ 001 = 10^{-9}$	nano	n
$0.000\ 000\ 000\ 001 = 10^{-12}$	pico	p
$0.000\ 000\ 000\ 000\ 001 = 10^{-15}$	femto	f
$0.000\ 000\ 000\ 000\ 000\ 001 = 10^{-18}$	atto	a

* To be avoided where possible.